HYPERSIGNAL-MACRO SERIES
HARDWARE REFERENCE GUIDE

(revision 4.0)
October, 2000

Signalogic, Inc.
9617 Wendell @ Skillman
Dallas, TX 75243
tel: 214-343-0069
fax: 214-343-0163

http:// www.signalogic.com
tech support: dsphelp@signalogic.com

TABLE OF CONTENTS

1.0 INTRODUCTION 1-1
2.0 HARDWARE CONSIDERATIONS 2-1
2.1 Compatibility Considerations 2-1
2.2 Performance Considerations 2-5
2.3 Sampled Data Resolution 2-6
2.4 Anti-Aliasing and Reconstruction Filters 2-7
2.5 Number of Analog I/O Channels 2-8
3.0 SPECIFIC HARDWARE 3-1
3.0.1 How to Read the Hardware Specifications 3-2
3.1 Analogic 3-3
3.1.1 Analogic 6400 Waveform Digitizer (No DSP Chip) 3-3
3.2 Ariel 3-5
3.2.1 Ariel DSP-16 (TMS320C25) 3-6
3.2.2 Ariel PC-C25 (TMS320C25) 3-8
3.2.3 Ariel DSP-56 (DSP56001) 3-10
3.2.4 Ariel PC-56 (DSP56001) 3-12
3.2.5 Ariel PC-56D (DSP56001) 3-14
3.2.6 Ariel DSP-32C (DSP32C) 3-17
3.2.7 Ariel PC-32C (DSP32C) 3-19
3.2.8 Ariel DSP-96 (DSP96002) 3-22
3.2.9 Ariel MP3210 (DSP3210) 3-24
3.2.10 Ariel DSP-C40 (TMS320C40) 3-26
3.3 Atlanta Signal Processors, Inc. (ASPI) 3-28
3.3.1 ASPI Banshee (TMS320C30) 3-28
3.4 AT&T 3-30
3.4.1 AT&T DSP32C Development Board 3-30
3.5 Communications, Automation, and Control (CAC) 3-32
3.5.1 CAC DSP32-PC (DSP32) 3-33
3.5.2 CAC DSP32C-AT (DSP32C) 3-34
3.5.3 CAC CAC-32C / AC5 (DSP32C) 3-36
3.5.4 CAC QUANTUMdsp (DSP3210) 3-38
3.6 Data Translation 3-40
3.6.1 Data Translation DT2821 (No DSP Chip) 3-40
3.6.2 Data Translation DT2823 (No DSP Chip) 3-42
3.7 Digital Audio Labs 3-44
3.7.1 Digital Audio Labs DRB1 (No DSP Chip) 3-44

3.7.2 Digital Audio Labs DRB2 (No DSP Chip) . . . 3-46
3.8 Domain Technologies . . . 3-47
3.8.1 Domain Technologies DSPCard (TMS320C31) . . . 3-47
3.9 DSP Research . . . 3-50
3.9.1 DSP Research Tiger-30 (TMS320C30) . . . 3-50
3.10 Innovative Integrataion . . . 3-53
3.10.1 Innovative Integraion ADC64 (TMS320C32) . . . 3-53
3.10.2 Innovative Integration M44 (TMS320C44) . . . 3-54
3.10.3 Innovative Integraion PC31 (TMS320C31) . . . 3-54
3.10.4 Innovative Integration PC32 (TMS320C32) . . . 3-59
3.10.5 Innovative Integration PC44 (TMS320C44) . . . 3-60
3.10.6 Innovative Integration PC50 (TMS320C50) . . . 3-60
3.10.7 Innovative Integration Modules . . . 3-64
3.11 Intelligent Instrumentation, Inc. (I^3) . . . 3-66
3.11.1 I^3 PCI20202C (TMS320C25) . . . 3-66
3.11.2 I^3 PCI20041C (No DSP Chip) . . . 3-70
3.11.3 I^3 ZPB32 (DSP32) . . . 3-77
3.11.4 I^3 ZPB34 (DSP32C) . . . 3-80
3.12 Loughborough Sound Images (LSI) / Spectrum Signal Processing . . . 3-84
3.12.1 LSI/Spectrum TMS320C25 System Board . . . 3-84
3.12.2 LSI/Spectrum DSP56001 System Board . . . 3-87
3.12.3 LSI/Spectrum DSP32C System Board . . . 3-89
3.12.4 LSI/Spectrum TMS320C30 System Board . . . 3-91
3.12.5 LSI/Spectrum ADSP-2101 System Board . . . 3-92
3.12.6 LSI/Spectrum ADSP-2100 System Board . . . 3-95
3.12.7 LSI/Spectrum DSP96002 System Board . . . 3-96
3.13 MTT Instrumentation . . . 3-100
3.13.1 DSP4100 System . . . 3-100
3.13.2 DSP4200 System . . . 3-103
3.14 Motorola . . . 3-105
3.14.1 Motorola ADS56000 Development System (DSP56001) . . . 3-105
3.15 National Instruments . . . 3-108
3.15.1 National AT-DSP2200 . . . 3-108
3.16 NEC . . . 3-110
3.16.1 NEC DDK-77P220 Development Board (77220) . . . 3-110
3.17 Signalogic . . . 3-111

3.17.1 Signalogic Sig32C-4 (DSP32C . . . 3-111
3.17.2 Signalogic Sig32C-8 (DSP32C) . . . 3-113
3.17.3 Signalogic SigBook (DSP5630x) . . . 3-116
3.17.4 Signalogic SigC31-4 (TMS320C31) . . . 3-117
3.17.5 Signalogic SigC32-2 (TMS320C32) . . . 3-117
3.17.6 Signalogic SigC44-PC/104 (TMS320C44) . . . 3-119
3.18 Sonitech . . . 3-122
3.18.1 Sonitech Spirit-30 (TMS320C30) . . . 3-122
3.19 Techron . . . 3-126
3.19.1 Techron TEF20HI (DSP56001) . . . 3-126
3.20 Texas Instruments . . . 3-128
3.20.1 Texas Instruments SWDS Board (TMS320C25) . . . 3-128
3.20.2 Texas Instruments EVM Board (TMS320C30) . . . 3-131
3.21 Turtle Beach Systems . . . 3-133
3.21.1 Turtle Beach Systems Tahiti (DSP56001) . . . 3-133
4.0 DSP SOURCE CODE INTERFACE . . . 4-1
4.0.1 Source Code License Agreement . . . 4-2
4.1 Texas Instruments TMS320C25 DSP Source Code Interface . . . 4-3
4.1.1 TMS320C25 Overview . . . 4-3
4.1.2 TMS320C25 Source and Support File List . . . 4-4
4.1.3 TMS320C25 Modification Process . . . 4-5
4.2 AT&T WE DSP32 DSP Source Code Interface . . . 4-7
4.2.1 WE DSP32 Overview . . . 4-7
4.2.2 WE DSP32 Source and Support File List . . . 4-7
4.2.3 WE DSP32 Modification Process . . . 4-7
4.3 Motorola DSP5600x DSP Source Code Interface . . . 4-9
4.3.1 DSP5600x Overview . . . 4-9
4.3.2 DSP5600x Source and Support File List . . . 4-10
4.3.3 DSP5600x Modification Process . . . 4-12
4.4 AT&T WE DSP32C DSP Source Code Interface . . . 4-13
4.4.1 WE DSP32C Overview . . . 4-13
4.4.2 WE DSP32C Source and Support File List . . . 4-14
4.4.3 WE DSP32C Modification Process . . . 4-15
4.5 Texas Instruments TMS320C3x DSP Source Code Interface . . . 4-18
4.5.1 TMS320C3x Overview . . . 4-18

4.5.2 TMS320C3x Source and Support File List 4-19
4.5.3 TMS320C3x Modification Process 4-20
4.6 NEC 77220 DSP Source Code Interface 4-23
4.6.1 77220 Overview 4-23
4.6.2 77220 Source and Support File List 4-24
4.6.3 77220 Modification Process 4-25
4.7 Analog Devices ADSP-210x DSP Source Code Interface 4-27
4.7.1 ADSP-210x Overview 4-27
4.7.2 ADSP-210x Source and Support File List 4-27
4.7.3 ADSP-210x Modification Process 4-29
4.8 Motorola DSP96002 DSP Source Code Interface 4-31
4.8.1 DSP96002 Overview 4-31
4.8.2 DSP96002 Source and Support File List 4-32
4.8.3 DSP96002 Modification Process 4-33
4.9 AT&T DSP3210 DSP Source Code Interface 4-35
4.9.1 DSP3210 Overview 4-35
4.9.2 DSP3210 Source and Support File List 4-36
4.9.3 DSP3210 Modification Process 4-37
4.10 Texas Instruments TMS320C4x DSP Source Code Interface 4-39
4.10.1 TMS320C4x Overview 4-39
4.10.2 TMS320C4x Source and Support File List 4-40
4.10.3 TMS320C4x Modification Process 4-41
4.11 Texas Instruments TMS320C5x DSP Source Code Interface 4-43
4.11.1 TMS320C5x Overview 4-43
4.11.2 TMS320C5x Source and Support File List 4-44
4.11.3 TMS320C5x Modification Process 4-45
Appendix A: WriteDSPVal Constants A-1

this page intentionally left blank

1.0 INTRODUCTION

Hypersignal supports several different manufacturers and models of DSP and analog acquisition hardware. A general characteristic shared by all supported hardware is the ability to input/output analog signals and convert them to/from digital form. In most cases the analog capability is integral to the hardware ("onboard"), allowing the analog signals to be connected directly to the backplane of the computer once the board is installed. In some cases the analog hardware resides outside the host PC (usually, but not always, in a separate, "noise-resistant" enclosure) and is connected digitally to a small interface card actually inside the host PC. Analog capabilities, such as number of channels, anti-aliasing filters, programmable gain and trigger, and speed of operation, vary greatly from one manufacturer to another. Another general characteristic shared by all supported hardware is the ability to transfer data to and from host memory over the host I/O bus. Digital capabilities, such as the method of data transfer and the presence or absence of a DSP (digital signal processing) chip also vary between manufacturers. Chapter 2 below, HARDWARE CONSIDERATIONS, attempts to create a basic framework in which both analog and digital differences between hardware can be discussed and compared.

Chapter 3, SPECIFIC HARDWARE, discusses installation, operation and specifications of individual boards. All hardware discussed in this manual are supported by Hypersignal.

Chapter 4, DSP SOURCE CODE INTERFACE, discusses modification of the source code that executes on those boards with a DSP chip. The power of this approach is enormous, since many functions in the high-level Hypersignal host software depend on the analog/DSP hardware for their necessary data and subsequent sequence of operations. *Modifying the low-level DSP source code can, in effect, change the way the entire system operates.*

The DSP Source Code Interface includes both library routines and also real-time, interrupt-driven structures that allow foreground and background processing in conjunction with sample interrupts. Initially, the source code produces standard Hypersignal operation, including instrumentation displays, continuous transfer to/from disk, and file-based acceleration, with no programming required. In other words, if you were to assemble and link the source code for a specific DSP chip as it was shipped, without any changes, you would end up with the distribution object file that is sent with every Hypersignal package. However, the source code could then be modified and/or added to in order to enhance the baseline Hypersignal operations or add new functions entirely. Changes can be as simple as interfacing to new analog converters or as complicated as simulating and testing complete algorithms by transferring data between the DSP board and Hypersignal software resources (including hard disk and video display) using pre-defined buffer-switching mechanisms and real-time interrupt structures.

2.0 HARDWARE CONSIDERATIONS

The concept is simple: put a board with analog input/output in your PC and turn it into a desktop data logger, oscilloscope, spectrum analyzer, or signal processor. And/or implement application-specific real-time algorithms with a powerful DSP chip inside your PC. Theoretically, the PC as host allows much greater potential than traditional stand-alone instruments and development systems, due to its well-integrated video display and mouse, hard disk, and tremendous supporting software. Practically, however, there are many hardware characteristics and limitations that must be considered in order to achieve the best possible analog and/or DSP performance in the PC environment.

As designers of real-time signal processing software, we have a unique perspective on the subject of analog/DSP boards for PCs. Because we have to write the software that not only makes these things work, but, indeed, takes full advantage of their features, we have come to know individual boards from over a dozen different manufacturers inside-out. Moreover, after 6 years and so many board integrations, we have gained a broad viewpoint, generalized knowledge, and an ability to recognize key features of hardware.

Sections 2.1 and 2.2 below, Compatibility Considerations and Performance Considerations, respectively, present discussions of issues that are of general importance when using plug-in DSP/analog hardware in a PC. They are strongly recommended reading, especially if you suspect a problem or performance that seems to be subpar.

2.1 Compatibility Considerations

Performance considerations and data transfer methods described in section 2.2 below mean nothing if the desired hardware is not compatible with the desired PC in the first place. There are two major areas of compatibility concern: 1) if a memory-mapped transfer scheme is used, and 2) PC bus speed (or internal bus speed).

MEMORY-MAPPED DATA TRANSFER GOTCHAS

This section is only applicable if your DSP/analog hardware uses a memory-mapped transfer scheme (see the individual hardware sections in chapter 3, under “Digital Specifications”, and section 2.2 below, Performance Considerations, under “Method and Speed of Data Transfer”).

Two 64 kbyte segments of the the PC’s “high memory”, between 640k and 1024k, have traditionally been reserved for application-specific plug-in boards and/or memory-resident software utilities. The starting address for these segments is usually D000:0000H and E000:0000H. However, over time, such a plethora of boards and utilities have been created which require significant amounts of this limited resource, that conflicts can easily arise on any given PC. Most DSP board manufacturers try to avoid using memory-mapped schemes, to save themselves the technical support headaches, but the temptation of somewhat better performance (and sometimes easier design) is strong, and there are still boards out there which use this technique. If you have DSP/analog hardware which uses the memory-mapped transfer mechanism (see the individual hardware sections in chapter 3, under “Digital Specifications”), here are some symptoms to watch out for after installing the hardware:

- hardware seems to run for a while and then freeze up; hardware gets through part of a function and then locks up
- data spikes or glitches are displayed in functions like the digital oscilloscope or spectrum analyzer
- another, unrelated board or previously working software utility suddenly develops a problem
- otherwise strange or intermittent system behavior that was not present before installation of the DSP/analog hardware

If one of the above symptoms seems to be occurring, you may have encountered a memory-mapped gotcha. If so, here are some things you can check:

1) The first thing you should be aware of is that if a line similar to

device = c:\dos\himem.sys

or

device = c:\windows\himem.sys

appears in your config.sys file, then any utility or program invoked afterwords could be loaded so as to conflict with memory needed by the DSP/analog hardware. This is because himem.sys makes it possible to load other programs in the high memory area mentioned above. If the HIMEM.SYS line is not present, you should add it. This line should appear before any other program or utility is invoked.

2) Look in your config.sys file for an "EMM386" line. You may have something like

device = C:\windows\emm386.exe

or

device = C:\dos\emm386.exe

followed by various memory usage specifications. The emm386 program controls PC access to high memory. It also controls access and usage of "expanded memory", which is an antiquated technique used for peripheral devices or software utilities which need a substantial amount of extra memory (i.e. more than .25 Mbyte). If this line is not in your config.sys file, you should add it. Then, to reserve access to your memory-mapped DSP/analog hardware, try adding the following to this line:

X=AAAA-BBBB NOEMS

Only add "NOEMS" if not already there; it must be after the "X=" command. If you are adding the EMM386 line for the first time, put a space after emm386.exe. The "X=" command specifies a certain section of high memory be reserved. The AAAA should be the starting address of the memory needed by your DSP/analog hardware, and the BBBB the last address needed. Some examples (partial list only):

Hardware	EMM386 line addition
Ariel DSP-16	X=D000-DFFF
Ariel DSP-3210	X=DF00-DFFF
Innvoative Integration PC31	X=DC00-DFFF
DSP Research TIGER 30	X=DC00-DFFF

NOTES: It is important the "EMM386" line appear immediately after the HIMEM.SYS line. This will ensure that not only does emm386.exe prevent other programs from being loaded into the reserved area, but emm386.exe itself does not load into the reserved area.

"NOEMS" specifies no expanded memory; if you think your system is using expanded memory, then the situation is more complicated. This would be the case if the emm386.exe line is already in your config.sys file and it does not contain the NOEMS or X= commands. In this case, you should not add NOEMS to the emm386.exe line. You will need to figure out which segment of high memory your expanded memory is using, and set up the DSP/analog hardware to avoid this segment. For example, if your expanded memory is using segment D000:0000, then you should change the base address on your DSP/analog hardware from Dxxx Exxx (leave the lower digits the same), and then add an X=Eaaa-Ebbb command to your emm386.exe line, following the guidelines explained above.

You can also add a "V" to the end of the emm386 line to cause the status and results of the emm386 program to be printed onscreen when your computer is booting up.

3) Look in your SYSTEM.INI file (found on \WINDOWS\SYSTEM subdirectory) for "EMMExclude=" lines, located in the "[386Enh]" section of this file. If this section exists, you should add an EMMExclude lines which matches up with the EMM386 command you have entered in your config.sys file. If, for example, your config.sys file contains an "X=D000-DFFF" command, then you would add an "EMMExclude=D000-DFFF" line in the EMM section. (Actually, Microsoft documentation indicates that "X=" commands in the "EMM386" line take priority over EMMExclude lines in the SYSTEM.INI file, but we think it always a good idea to be extra safe with DOS-Windows interaction).

4) Look for other boards in your system which may be using the same high memory segment. Typical culprits include network cards (which almost always require a small section of either the D000:0000 or E000:0000 segments), some high-performance video cards, other DSP/analog boards or sound boards (of course!), and non-standard peripheral cards (anything off the beaten track). If you find one, first simply try removing it before doing anything else. If the DSP/analog hardware problems clear up,

then one of the contenders will need to have its base memory address changed. If you change the DSP/analog hardware base address, be sure to inform Hypersignal (see the individual hardware sections in chapter 3).

5) Look for software utilities or memory-resident programs which might be using high memory. You can do this by using the "mem" or "mem /s" commands from the DOS command line. Occasionally, such programs can be so ill-behaved they even use areas reserved by the emm386.exe program (see above). In such cases, you have no choice but to temporarily "REM-out" these utilities one-by-one from your config.sys or autoexec.bat files until you find out what is causing the problem. Sometimes, you can simply change the order in which these utilities are invoked to make a problem go away—loading ill-behaved programs later increases the chance hey will use areas reserved by emm386.exe, since other (well-behaved) programs have loaded themselves with due respect to areas reserved by emm386.exe.

NOTES: Changing the order of utilities and programs in your config.sys and autoexec.bat files can be a risky business. Change the order only of suspected problem-causers. These should not be well known utilities such as smartdrive, share, doskey, dosshell, himem.sys, ansi.sys, etc; they are much more likely to be programs supplied by third-party manufacturers, or having to do in some way with peripheral cards you have added to your system.

Below are some excerpts from files gathered from the Microsoft Knowledge Base on the CompuServe network (all excerpts COPYRIGHT MICROSOFT 1994). These files should indicate the same information as presented above, but we've included them because they offer some extra detail which could prove useful in certain situations.

EMM386 Troubleshooting and Technical Information

Copyright Microsoft, 1994
Title: EMM386 Troubleshooting and Technical Information
Document Number: Q78557
Publ Date: 17-SEP-1994
Product Name: Microsoft Disk Operating System
Product Version: 5.00 5.00a 6.00 6.20 6.21 6.22
Operating System: MS-DOS

If EMM386.EXE locks up the computer, follow these troubleshooting steps:

1. If the DEVICE=EMM386.EXE in the CONFIG.SYS file contains the HIGHSCAN parameter, remove HIGHSCAN from the command, save the CONFIG.SYS file, and restart your computer. (HIGHSCAN cannot be used on some computers.)

2. Start EMM386.EXE with the exclude option.

EMM386.EXE may have incorrectly identified an area being used by the system as a "hole" (a region that can be used as a UMB or an EMS page frame). As a result, EMM386.EXE overwrites a portion of memory being used by a hardware adapter in your computer.

By excluding addresses in the range A000-EFFF, EMM386.EXE does not use any part of the excluded region for a UMB or EMS page frame. Identifying the correct region(s) to exclude requires experimentation. Start by excluding a large region and then reducing the size of the region. For example:

DEVICE=EMM386.EXE NOEMS X=A000-EFFF

DEVICE=EMM386.EXE NOEMS X=C000-DFFF

DEVICE=EMM386.EXE NOEMS X=C800-CFFF

You can use multiple exclusions on the EMM386.EXE device line. For example:

DEVICE=EMM386.EXE NOEMS X=C000-C7FF X=E000-EFFF

Refer to the documentation included with your hardware device(s) for information on the upper memory addresses the adapter card uses. Examples of devices which use upper memory addresses include network cards, SCSI adapters, video cards, and scanners.

How EMM386 and Windows 3.1 Relate (Include/Exclude Precedence)

Copyright Microsoft 1994
Title: How EMM386 and Windows 3.1 Relate
Document Number: Q82711
Publ Date: 13-OCT-1994
Product Name: Microsoft Windows 3.x Retail Product
Product Version: 3.10 3.11
Operating System: WINDOWS

An X= parameter added to the EMM386.EXE device line is directly adopted by Windows. Any EMMInclude= or EMMExclude= entry is overridden by any X= parameter on the EMM386.EXE device line in the CONFIG.SYS file. For example, the following line in your CONFIG.SYS takes precedence ove any conflicting entry:

device=c:\windows\emm386.exe X=C000-CFFF noems

When Windows executes, it takes this information before it checks the SYSTEM.INI file. Even if the SYSTEM.INI file contains a contradictory line, for example:

EMMInclude=C000-CFFF

Windows keeps the EMM386.EXE parameter.

How to Troubleshoot Hardware Conflicts

Copyright Microsoft 1994
Title: How to Troubleshoot Hardware Conflicts
Document Number: Q90245
Publ Date: 11-OCT-1994
Product Name: Microsoft Windows 3.x Retail Product
Product Version: 3.10
Operating System: WINDOWS

Hardware conflicts are difficult to diagnose because they manifest themselves in different ways depending on the type of network card and what they are conflicting with (disk driver, sound board, and so on). Typical hardware symptoms include:

1. Failure of the network card device driver to load; Error 5733 when the NET START command is used, or the network card fails to bind.

2. Inconsistent network behavior; the system stops responding (hangs) sometimes, orerrors occur copying large files over the network.

3. Error 53: trying to connect to servers. (You cannot see anything on the network.)

If you suspect some type of conflict, try one of the following:

- Switch the network card to a different interrupt.
- Check for memory conflicts between the network card and other devices in the system. Some network cards use an of upper memory. If the area of memory, the card used is configurable. See the documentation for the network card. You can determine which parts of upper memory are being used by other devices by doing the following:

  1. Remove the network drivers and EMM386.EXE from CONFIG.SYS file.

  2. Restart the computer.

  3. Run Microsoft Diagnostics (MSD.EXE).

  4. Look at the memory map. It should show any upper memory being used. If the net card is trying to use the same area as another device, reconfigure the network card to use a different memory area.
- Make sure the memory range for the network card (if it uses one) is excluded with EMM386.EXE if it is being used, or in the [386enh] section of SYSTEM.INI file if EMM386.EXE in not being used.

HOST PC INTERNAL BUS SPEED

Strictly speaking, DSP/analog hardware is not expected to operate in a host PC with internal bus speed greater than 8.33 MHz, which was the original IBM-AT standard. However, over time, a situation has evolved where many boards of different types can operate at higher bus speeds to provide better performance. And so, computer manufacturers have offered BIOS firmware options or motherboard switches which allow different internal bus speeds to be selected. Typically, one can select an internal bus speed that is the result of dividing the master PC clock speed by some integer. For example, on a 66 MHz 486 machine, options may be available for 13.2, 11, 9.43, and 8.25 MHz bus speeds. Often times, a machine has been configured by the manufacturer or system integrator to utilize hard disk controller, video card, etc. which can accept a higher speed, in order to provide better performance when delivered to the customer. Unfortunately, not all DSP/analog boards can tolerate increases above the 8.33 MHz mark. Cases where testing at Signalogic has shown this to be the case are indicated as highlighted notes in the individual hardware sections in chapter 3. In such cases, you may need to run the BIOS configuration or setup program and look for an option to reduce the host PC internal bus speed.

2.2 Performance Considerations

In the paragraphs below, we will present a general discussion of many of the tradeoffs which might be considered when operating analog/DSP hardware with a PC. Note that we always discuss hardware from the viewpoint of being used *in conjunction with high-performance software such as Hypersignal*, and never from a stand-alone or minimal-software point of view. This reference guide is by no means intended to be a complete text for fully understanding all analog/DSP hardware—but it is a start. Please call us and ask to speak to an applications engineer when you have further questions.

SAMPLING RATE / DSP PROCESSOR CONSIDERATIONS

The first item of importance is the obvious: how fast must the hardware sample data? That is, how many times per second must the hardware go out into the real world and measure/set a physical quantity (Voltage) at its input/output?

With state-of-the-art in PC analog/DSP boards today, sampling rates from a few milliHertz to a few MegaHertz are possible. However, one must consider the options carefully, as tradeoffs with respect to sampling frequency lurk everywhere. Generally, hardware with rates between 150 Hz and 50 kHz are quite common, and can be obtained without any trouble. But below or above this range, problems can appear. First, if you are in the minority who needs *very slow* sampling rates, instead of very fast, there is only one basic problem. Although very slow sampling inherently represents no technical risk, there are few boards available (with or without DSPs) which sample below 70 Hz. This is because most manufacturers feel a marketing need for faster sampling rates, and often, the technology used for fast sampling and that used for slow sampling are incompatible. Putting both on a board may be expensive, from the manufacturer's viewpoint. Thus, finding a board is really the only problem when trying to get down into the milliHertz zone. If you are in the majority for whom speed is the big concern, the rest of this section on sampling rate is pertinent.

One aspect which needs to be considered at the same time as "speed" is whether the board has a DSP chip (or more than one DSP chip). Generally speaking, most boards *without* DSPs available today are capable of maximum sampling rates ranging from 50 kHz up to a few MHz. Most boards *with* DSPs available today are capable of maximum sampling rates ranging from about 50 kHz to 150 kHz (some can go faster, but this is misleading because today's current state-of-the-art DSPs tend to become much less effective for doing work above 200 kHz—but more about this later).

Of course, if *any sort of processing* must be performed on the data in real-time, a DSP is essential. *Do not* think that you can write optimized 80x86 assembly language programs to do some "simple processing" to real-time data from a board with no DSP, like vector transformations, or even adding offsets. The 80286 and 80386 have intrinsic, built-in instructions which allow transfer of data from memory-to-memory and memory-to-I/O without handling individual points. These instructions are what allows fast PCs to continuously transfer data to/from disk at rates up to 250 kHz. If you write 80x86 code which requires processing of each individual point, you will preclude the use of these instructions and drastically cut your maximum rate of real-time data transfer.

On the other hand, if sampling speed is the most important consideration, then a DSP may be out of the question. The fastest, widely-used, general-purpose DSP chips on the market today allow about two (2) floating-point operations every 60 nsec. If you want to sample at 400 kHz, you will have 2.5 μsec available for processing between samples. This translates to about 84 floating-point instructions. Of course, *most* of the instructions you will be using, especially the frequently-occurring, truly useful ones such as data transfer, comparisons, branches, etc. can't be arbitrarily combined into pairs without a migraine program-

ming headache, so really you will have about 50 or so instructions to play with between sample interrupts. This might be just enough to get the data, buffer it, check for a threshold, maybe run a 5-tap FIR filter, and communicate with the host PC every so often. Thus, for high-rate projects, unless real-time processing is required, you may want to consider an analog-only board to get/view the data, then apply your processing and algorithms after-the-fact, on a file-basis (simulation).

The sampling rate question can lead to all sorts of other tradeoffs besides use of a DSP, such as data resolution (number of bits per sample) and maximum number of input/output channels, which will be discussed as separate itmes below. One sampling rate tradeoff, limitation on the amount of data to be processed continuously, is worth further discussion.

If you need large amounts of data without gaps (continuous operation) at sampling rates above 50 kHz, then you will be limited more by your system capabilities and the transfer method of the board you choose, than by the maximum sampling rate specifications for individual boards. If your maximum sampling rates are below 50 kHz, or you can work with "snapshots" of your data, displayed on your PC video and/or captured to disk, then your system capabilities and the finer nuances of your board design should be transparent. Snapshots of various sizes can be displayed by the Hypersignal instrumentation functions (Digital Oscilloscope and Spectrum Analzyer). Another Hypersignal function, Analog Conversion, is designed totally for speed and simply transfers data to and from disk.

METHOD AND SPEED OF DATA TRANSFER

Generally, analog-only boards with no DSP chip use a direct memory access (DMA) interface to transfer data over the host internal bus to/from host memory. Very few boards with DSPs also implement DMA operation—the vast majority use a method called "programmed I/O", where the host accesses memory buffers on the board solely through I/O space (using "autoincrement" data ports for repetitive access). This results mainly from the fact that DSP boards need large amounts of random-access memory anyway (for the DSP chip itself). DMA is not good at random-access, and implementing DMA becomes basically redundant since the DSP and the host can now communicate through shared memory on the board.

Well-designed boards which support the programmed I/O method can transfer data nearly as fast as the DMA method, since the transfer instructions are intrinsic to the 80x86 and essentially operate much like DMA. Note that we say "well-designed": questions such as whether the board memory is fully dual-access and whether the board can keep up with the programmed I/O rate become important. To further complicate matters, the amount of shared memory on the DSP board is also important, because this is where data must exist before programmed I/O transfer can take place with the host PC. To achieve rates above 50 kHz, there should be enough data memory for at least two 12-16 kword buffers on the board, otherwise the head seek delays, file fragmenting, bad tracks, etc. which occur on a regular basis with disk operation could cause discontinuities in the data. Because of the numerous questions concerning board design when considering high sample rates, all boards supported by Hypersignal software and covered in the individual board sections have software "PERFORMANCE SPECIFICATIONS" figures, in addition to the usual hardware specifications, listed for each board. There are values listed for continuous streaming to hard disk and maximum instrumentation display operation.

Note that regardless of method, the host internal bus imposes an upper limit on transfer speed since all boards reside on this bus. While this may change with the advent of 32-bit buses (e.g., EISA), it currently means that *continuous* streaming operation (gap-free data) to disk above about 250 kHz is generally not possible. Because this results from a system bus constraint, the upper limit holds regardless of whether the data destination is hard disk or RAM disk. In addition, to achieve hard disk rates above 200 kHz, your system hard disk must be *extremely* fast. With snapshot operation, where a picture is taken every so often and data is skipped between pictures, it is usually possible to operate up to the maximum rate of the hardware. For example, with a Digital Oscilloscope function, you could see a few msec of 1 MHz data at a time on the video display, and/or store it to disk for further processing. To operate the instrumentation displays completely in real-time, without missing any "frames" of data, usually becomes problematic above 10-15 kHz, due to the amount of time it takes to draw waveforms in video memory, the amount of display area needed by the current data amplitude and frequency, etc. However, at higher rates you see the correct "instantaneous" response, since the analog/DSP hardware is not actually skipping any data, as is the video display.

2.3 Sampled Data Resolution

It is generally true that as the sampling rate goes up—especially beyond, say, 250 kHz—the number of bits in the digital input/output goes down. Rarely does one see hardware above 500 kHz with 16-bit resolution, or above 1 MHz with resolution above 12 bits. Besides speed, another big reason to go to a lower number of bits is cost: many times the cost of a hardware setup is significantly less for 12-bits, especially if the board contains multiple channels.

A more subtle question concerning data resolution is whether the board's noise floor is really good enough to support a higher number of bits. In the specific hardware sections, under the "noise floor" specification for each board, we list the dB down from full-scale input to a "peak-hold" noise floor for the Spectrum Analyzer function with the board's input shorted to ground (20 kHz sampling rate, 1024pt. FFT with no zero-fill, Blackman window, unless otherwise noted).

2.4 Anti-Aliasing and Reconstruction Filters

Anti-aliasing filters are a very important consideration, even if many times ignored by some manufacturers. Until recently, most boards had no provision for anti-aliasing filters on input analog data or reconstruction filters on output analog data, forcing the use of external filter units, but this has changed substantially in the last few years.

If the hardware in use has no anti-aliasing filters, there are two essential ways to counter this situation. First, active or passive RC network filter "blocks" can be purchased separately and inserted after the signal source and before the acquisition hardware. Such filter blocks are external to the rest of your system, require their own power supply if active, and are usually a high-order Butterworth, Chebyshev, elliptic, or Bessel type design. They also vary a great deal in expense, depending on their "goodness": filter order, phase, and resistance to environmental variations. A second, not so obvious, method is to sample fast enough to take advantage of any natural anti-alias protection built into your transducer. For example, in an acoustic application the transducer would be a microphone, which might have a natural bandwidth of about 20 kHz, with a frequency response which rolls off beyond that point. In such a case, sampling at around 40 to 45 kHz would guarantee no significant aliasing error in the acquired signal. If lower sampling rates were desired in the data for analysis purposes, a digital lowpass filter could be applied to the acquired data, followed by "decimation" to convert the initial sampling rate to the lower, desired value (Hypersignal performs sampling rate conversion with ease). Note that the application of a digital lowpass filter is not valid to *prevent* anti-aliasing, only to further filter already non-aliased data.

Hardware with anti-aliasing filters generally falls into one of three categories: 1) Fixed cutoff-frequency filters (board is usually intended for a certain user, or market segment). 2) Resistor-programmable filters, where you change resistor values depending on your sampling rate. 3) Boards with separate "signal processing ICs" which communicate digitally with a DSP chip.

Fixed cutoff-frequency filters are usually very good filters with more or less linear phase response, very high-order (steep filter response roll-off), and sinx/x correction on the output. However, if the cutoff-frequency does not agree with your sampling rate, then you have a problem. Resistor-programmable filters, where you change resistor values depending on your sampling rate, probably represent the best price/performance tradeoff—the only time this approach would not be acceptable is if the filter order wasn't high enough, or you expected to change sampling rates so often as to make changing resistor values impractical. If you choose a board without filters, you may pay significantly less, but you may find it necessary to bandlimit your analog data using external filters anyway. Note that simply applying a digital filter won't work unless you oversample, which might mean too high a sampling rate for continuous operation or instrumentation operation (see SAMPLING RATE discussion, above). In the specification data for each board listed in chapter 3, we list the type of filters and the filter order.

Recently, there have been two new trends concerning anti-aliasing filters for boards with DSP chips: the use of AICs (analog interface chips) for low-end boards and the use of sigma-delta A/D converters for high-end boards.

The AIC chip is basically a one-channel monolithic A/D converter and D/A converter, with a complete set of filters, which communicates digital data with the DSP via a serial bit stream. AICs use traditional successive-approximation converter technology, and are limited to about a 20-40 kHz maximum sampling rate. Because everything is on one chip which connects directly to the DSP, the complexity and cost of the board is reduced. Also, the AICs are programmable, allowing sampling rate, filter cutoff-frequency, input gain, and other on/off switches (filter enable/disable, loopback enable/disable, etc.) to be controlled via the DSP chip and, in turn, the high-level software.

The second new trend is the sigma-delta A/D converter. You may well have already heard something about this new breed of A/D converter. The sigma-delta concept generally means an A/D converter which greatly oversamples input (for example at a 64x or 128x rate), assuming that there is no significant noise energy present in the analog signal at frequencies beyond this high sampling rate, then applies an on-chip digital filter and decimation process to produce a sampled data stream at the desired rate. The digital filter is of the FIR (finite impulse response) variety, guaranteeing exactly linear phase. Dynamic range specifications for current sigma-delta parts generally exceed their traditional successive-approximation A/D converter counterparts by 2-3 dB, but more importantly, the sigma-delta parts require no external anti-aliasing circuitry, greatly reducing the complexity of the hardware design. Because the sampling rate is based on a master clock input, the anti-aliasing filters automatically track any change in the clock rate, allowing truly programmable sampling rates. Moreover, depending on the length of the digital fil-

ter, the anti-aliasing cutoff can be made so close to a “brickwall” as to not be worth arguing. Various boards offer various possibilities for generating different master clock rates to the sigma-delta chips.

You may be saying “this sounds like a free ride”. Well, it isn’t. One penalty paid when using sigma-delta technology is *delay* due to the long digital filters. For example, with current technology based on 64x oversampling, a 10 kHz sampling rate might involve up to a 3.5 millisecond delay between actual physical signals and the time when DSP sees the corresponding samples. For control and other feedback-based systems, this can be a serious problem. Another penalty paid when using the sigma-delta technology is the inability to sample slowly. As an illustration, consider a factory automation or seismic system that needed a 1 Hz sampling rate. This would result in only a 64 Hz oversampling rate, which would obviously allow for all sorts of common, everyday system noise to be present in the signal, again necessitating an external analog filter. One answer to this problem is to sample much faster and then implement the decimation process described above to achieve the desired effective rate.

Monolithic parts with delta-sigma D/A conversion are also available, reducing analog board design complexity and increasing analog performance and reliability even further. Essentially, the delta-sigma technology works in reverse: the signal is first “stretched” or interpolated by some factor, thus increasing its sampling rate beyond the noise possibilities, and then filtered.

2.5 Number of Analog I/O Channels

The number of channels on individual boards varies greatly; the way in which manufacturers provide additional channels varies even more. For DSP boards, with only one exception in currently supported hardware (a Texas Instruments TMS320C25-based system), a peripheral analog I/O board must be connected to the DSP board to get more than two channels. Peripheral analog I/O boards are usually connected by a vendor-defined, parallel-interface connector on the DSP board. Although probably destined to change shortly, it remains a strange reality that most non-DSP boards are capable of multichannel (usually 4 to 8) operation while most DSP boards are limited to two. This situation probably owes 20% to the greater amount of board real-estate needed by the DSP, its memory, and its other supporting circuitry, and 80% to fact that many DSP professionals have in the past been doing mostly one and two channel applications because the DSP chips weren’t initially that powerful. Of course, this is changing—not only have the advent of AICs and sigma-delta converters (see ANTI-ALIASING discussion, above) greatly reduced the amount of board area needed per channel, but DSPs become ever-faster, constantly overtaking more multichannel applications.

When considering the maximum sampling rate for multichannel operation, be very careful. Generally, if specifications listed for a specific board are for one channel then you must divide this value proportionally depending on the number of channels to be added. If the specifications are listed for two channels then for disk streaming this is still true, but for instrumentation operation, additional channels may have minimal impact (up to a point). In a few cases only, the actual *hardware* maximum sampling frequency specification must also be divided by the number of input channels (indicated in the specifications by a “-1CH” after the value).

this page intentionally left blank

3.0 SPECIFIC HARDWARE

Two basic categories of hardware are described in this chapter sections: boards with DSP chips and boards without. Generally, the boards without use a DMA (direct memory access) transfer method, and the boards with use a shared memory "programmed I/O" transfer method. See section 2.2 above, Performance Considerations, for a detailed discussion of these two types of hardware.

In the indivdual hardware sections below, the following main categories of information are provided:

1) Basic Descriptive and Installation Details

2) Hardware Ratings and Software Throughput Specifications

3) Onboard Memory Usage

4) Additional Notes About Hypersignal Software Operation

BASIC DESCRIPTIVE AND INSTALLATION DETAILS

The Hypersignal software menu designator for the board, the basic configuration of the board (general speed, whether analog I/O is external or not, etc.), and special installation instructions (jumper lists, hardware errors that must be corrected prior to operation, host PC I/O bus speed limitations, etc.) are covered.

HARDWARE RATINGS AND SOFTWARE THROUGHPUT SPECIFICATIONS

A list of specifications, containing both absolute hardware ratings and actual throughput ratings when used in conjunction with Hypersignal, is presented for each board. Specifications also include information on standard and maximum memory configurations and I/O bus transfer methods. See the section "How To Read the Hardware Specifications" below for a list of abbreviations and more explanation about individual specifications.

ONBOARD MEMORY USAGE

General usage of the memory provided on the analog/DSP board (if any) is discussed, along with some idea as to whether a memory upgrade would be worthwhile. When first initializing analog/DSP hardware, Hypersignal first sizes the total memory area(s) on the board. Thus, Hypersignal automatically detects and utilizes any memory expansion/upgrades added to the hardware after initial purchase.

NOTE: For hardware with DSP capability, a very important point should be made regarding the amount of onboard memory. Generally, Hypersignal will provide greater functionality if the DSP board in use has a large amount of data memory. With more memory Hypersignal can use larger buffers for disk transfers to increase continuous acquisition rates, view and capture more data to file in the instrumentation displays, accelerate convolutions, and perform larger FFT size operations. What is a large amount of data memory? On the order of 64 kword, where a word is the natural wordlength of the DSP chip. And you must be careful here: if the board has separate data and *program* memory, the program memory may not be of use (see the individual board sections). Many boards are currently shipped with 16 kword or less as standard data memory size, which means that a memory upgrade purchase may activate some functions such as convolution/correlation acceleration and improve others, such as speed of continuous disk transfer.

ADDITIONAL NOTES ABOUT HYPERSIGNAL SOFTWARE OPERATION

Additional notes about the use of particular hardware with Hypersignal, especially data entry in software menus and individual menu fields, are discussed. For example, where a standard Hypersignal menu might offer an "external trigger" selection, a particular board might not support this feature. Other examples include features found on a limited number of boards, such as channel list entries for multichannel boards, and gain entry for boards with programmable gain hardware circuitry.

3.0.1 How to Read the Hardware Specifications

In the sections on specific hardware in this chapter, there are specifications listed for each board currently supported by Hypersignal-Macro DSP software. Specifications for each board is divided into 4 categories: 1) PERFORMANCE, 2) DIGITAL, 3) A/D, and 4) D/A.

Categories 2-4 were initially obtained directly from the board manufacturer's Users Manuals and data sheets, and have in most cases either been verified by testing or modified as necessary after testing of the hardware. Category 2, DIGITAL SPECIFICATIONS, always lists memory sizes in natural word-lengths of the DSP device used (e.g. 32-bits for floating-point DSPs).

Category 1, PERFORMANCE SPECIFICATIONS, results from integrated testing of the board in conjunction with Hypersignal software. The machine used for this testing was an AT-class 33 MHz 386 with approximately 18 msec avg. access time hard disk and 11 MHz internal bus. The "1 channel Hard Disk Record" test refers to the Analog Conversion function in Hypersignal software, and simply measures the maximum rate the system can stream continuous A/D data to hard disk. The "Max. N-chn Spectrum Analyzer Rate" refers to the Spectrum Analyzer function in Hypersignal software, and measures the maximum sampling rate at which the Spectrum Analyzer function will operate at before it produces invalid results, FFT overflow, or simply "locks up". This maximum rate is a complicated value interdependent on the speed of the DSP chip, the speed of the host internal bus, and the type of data transfer mechanism used on the board (note that the speed of the host computer video display makes no difference). The "1024 pt FFT Accel. Disk File Rate" test refers to an accelerated FFT function, measured over 500 1024 pt. frames (no zero-filling), read in from hard disk, processed by the hardware, and written back to hard disk. The "Spectrum Analyzer Noise Floor" test refers to the Spectrum Analyzer function in Hypersignal software, and is performed under the following conditions:

1) 50 kHz sampling rate (or maximum of hardware, if less)

2) single-channel displayed (log magnitude format)

3) no averaging; measurement is essentially "peak-hold", monitoring the highest spurious level from dc to 25 kHz

4) analog input shorted to ground (or to V- input if differential input), or left unconnected—whichever produces lower spurious results

5) 1024 pt. FFT, 512 pt. framesize, Blackman window

Due to the wide variance in different machines and "clones", even those with ostensibly identical characteristics, you may encounter discrepancies with the figures depending on the machine.

While the utmost care and effort has been made to obtain accurate measurements, neither Hyperception nor Signalogic can guarantee their accuracy, and are not responsible for errors and omissions.

Below is a legend for the abbreviations and acronyms used in the hardware specifications:

AIC	analog interface circuit
BESSELn	Bessel anti-alias or reconstruction filter type, followed by value indicating order of filter
BUTTERn	Butterworth anti-alias or reconstruction filter type, followed by value indicating order of filter
CODEC	coder/decoder (normally refers to μ-Law or A-Law log companding)
CHEBYn	Cheybshev anti-alias or reconstruction filter type, followed by value indicating order of filter
DI	differential-ended analog input
DMA	direct-memory-access transfer
DS	delta-sigma D/A converter
ELLIPn	elliptic anti-alias or reconstruction filter type, followed by value indicating order of filter
Fc=nn	fixed cutoff-frequency, usually followed by a value in kHz
Jnn	jumper with nn different positions
JB	jumper-bypassable
Hz	Hertz (samples/sec)
kHz	kiloHertz (10^3 samples/sec)
MM	maximum memory configuration
msec	millisecond (10^{-6} second)
NA	not applicable
nsec	nanosecond (10^{-9} second)
OB	offset binary

PA	potentiometer-adjustable (external knob)
Pnn	programmable via software, followed by number indicating number of discrete values possible
PCM	pulse code modulation—sending values by a series of pulses with different spacing (binary transmission)
Prog. I/O	programmed I/O transfer (see SAMPLING RATE discussion previous)
RP	programmable via resistor jumper or block (not software)
SD	sigma-delta A/D converter
SI	single-ended analog input
S/H	sample-and-hold
SWSn	Switch-selectable, n positions
TA	trimpot-adjustable
THD	total harmonic distortion
UA	figure unavailable
1CH	indicates maximum sampling rate must be divided by number of channels in use

In the DIGITAL SPECIFICATIONS section, "P" and "D" stand for program memory and data memory in cases where the DSP device uses a Harvard architecture (separate memory buses). In cases where the DSP device is Von Neumann (single bus), no designator is used. When no designator is used, the memory section *may be both instruction and data.* In cases of both program memory *and* split data memory, "X" and "Y" are used to indicate the separate data memory areas.

3.1 Analogic

3.1.1 Analogic 6400 Waveform Digitizer (No DSP Chip)

NOTE: Even though this board has no DSP chip, Spectrum Analyzer and DSP Accelerator operation can still be achieved via a separate, inexpensive "DSP Coprocessor" board. See section 3.5.1, below, for a description of such a board. See the product catalog for pricing information and further details.

Basic Descriptive and Installation Details

Use a hardware designator of 'AT6400' in the Hypersignal software System Configuration menu (except for DSP ACCELERATOR field). It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The 6400 Waveform Digitizer is a 2, 4, or 8-channel 14-bit simultaneous sample-and-hold system with extremely flexible and powerful programmable *analog* signal processing features, and a maximum single-channel sampling rate of 1 MHz. The Waveform Digitizer is an external unit that interfaces to the PC with a 1/2 length AT-style board with 16-bit ISA bus interface.

Operation of the 6400 Waveform Digitizer requires a memory-resident driver created and maintained by Analogic, which is unlike all other hardware currently supported by Hypersignal. The "TR.SYS" memory-resident device driver must be loaded in memory prior to Hypersignal operation. This should be done by adding the line

device = tr.sys

to your system's "CONFIG.SYS" file on the root directory. This will instruct your system to install "TR.SYS" in memory when your system first boots up. This implies that control of the I/O base address for the 6400 host interface board must be exercised using Analogic software that affects the "TR.SYS" device driver, and not through Hypersignal.

NOTES: We have found that maximum sampling rates for continuous (streaming) operation to disk with this system improve dramatically with increased I/O bus speed. We suggest 10 MHz, or even 11-12 MHz, if your system offers control over the bus speed.

After initially powering the system up, you may encounter an error message ("...unable to initialize 6400...") when first trying to execute a function using the 6400. Turn the 6400 off, wait 5 seconds, and turn it back on. This should allow the 6400 "TR.SYS" resident driver to initialize properly, if it has not already done so.

Input and output offset entries in the Analog Conversion function are active (enter in range -8191 to 8191 for 14-bit A/D) and applied digitally, via 80x86 assembly language. For optimum speed, use offset values of zero.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	None
Data Format, Wordlength:	None
1 channel Hard Disk Record:	150 kHz
1 channel Hard Disk Playback:	N/A
Max. 1-chn. Spectrum Analyzer Rate:	1 MHz
1024 pt FFT Accel. Disk File Rate:	N/A
Spectrum Analyzer Noise Floor:	79 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	N/A
Maximum Ext. Mem. Config:	N/A
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	DMA

A/D SPECIFICATIONS

Number of linear PCM channels:	1/2/4/8
Codec channels:	none
Expandable:	yes
Simultaneous S/H	yes
Typical Dynamic Range:	UA
THD:	.01 %
Voltage Range:	±.128V to ±8.192V; P4
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	1 MHz - 1CH
Min. Sampling Rate:	0
External TTL-Level Trigger:	yes
Programmable Level Trigger:	yes

D/A SPECIFICATIONS

Number of linear PCM channels:	N/A
Codec channels:	N/A
Expandable:	N/A
Simultaneous S/H:	N/A
Typical Dynamic Range:	N/A
THD:	N/A
Voltage Range:	N/A
Output Impedance:	N/A
Reconstruction Filters:	N/A
Sin(x)/x Filters:	N/A
Programmable Sampling Rate:	N/A
External Sample Clock:	N/A
Max. Sampling Rate:	N/A
Min. Sampling Rate:	N/A
External TTL-Level Trigger:	N/A

Onboard Memory Usage

There exists considerable onboard memory inside the 6400 (which allows for "negative-trigger" operation), but Hypersignal has no access to this memory—only the "TR.SYS" device driver accesses the interface board and 6400 directly.

Additional Notes About Hypersignal Software Operation

There are several significant deviations from standard Hypersignal operation for the Analogic 6400.

1) The 6400 has no D/A capability. The Analog Conversion menu is presented differently to account for this fact; also, the DSP ACCELERATOR field in the System Configuration menu takes over as the default D/A playback hardware in the Waveform Display/Edit and 2-D Spectrograph functions, allowing a second board to perform the D/A function if present in the system.

2) Channels are "one-based" instead of "zero-based"; i.e. in the instrumentation functions you must enter 'A1' for the first channel instead of 'A0'. In the Analog Conversion function the CHANNEL LIST field must start with a '1' entry.

3) In the Analog Conversion function entries in the CHANNEL LIST may be followed by an 'A' for AC coupling, a 'D' for DC coupling, and a 'C' for calibration. No entry assumes AC coupling. For example, an entry of '1,2D,3A,4C' would specify 4-channel operation, with AC coupling on channel 1, DC coupling on channel 2, AC coupling on channel 3, and calibration on channel 4.

4) Multichannel operation must always be in incremental groups of two: A1 and A2, A1-A4, etc.

5) The Analog Conversion INPUT RANGE LIST and instrumentation INPUT RANGE fields are fully operational and should be entered in Volts. Values of 8.192, 4.096, 2.048, 1.024, .512, .256, and .128 Volts are valid. The 6400 has programmable gain registers that are controlled by these fields, affecting signal gain before the actual A/D conversion process. In addition, the instrumentation function "software gain" controls are also active—in this case, digital scaling is being performed by fast 80x86 assembly language routines on recently-converted digital data. In the Analog Conversion function, you may assign gain values *independently* to channels currently in use by entering values of 8.192 (±8.192V full-scale), 4.096, 2.048, 1.024, .512, .256, or .128. Values separated by commas correspond to individual CHANNEL LIST entries. In the Digital Oscillscope and Spectrum Analyzer functions the same values can be used, but only one entered value is allowed, and it applies to all channels.

6) Unlike most other hardware with no DSP capability, the 6400 has programmable level comparators that can be used for triggering. The Analog Conversion TRIGGER LEVEL LIST field and instrumentation trigger controls are fully operational and operated in units of Volts.

7) The Analog Conversion TRIG MODE,DELAY field should be entered in the form of 'X,nnn', where X can be 'A' for Auto, 'N' for Normal, and 'F' for Freerun trigger operation. Nnn should be a trigger delay value in µsec, and can be either negative or positive. A negative trigger delay would cause the hardware to retain the specified amount of data prior to the trigger event in order to provide a "pre-history". In the instrumentation displays, trigger level values of zero (or very near zero) set the 6400 trigger mode to freerun, while non-zero values set the mode to Normal. A trigger delay cannot be specified in the instrumentation functions.

8) Unlike most hardware, the TRIGGER CHANNEL LIST field in the Analog Conversion function allows "nested" trigger conditions. In other words, if you specify a TRIGGER CHANNEL LIST of '1,2', then both channels 1 and 2 must meet the criteria specified in the TRIG MODE,DELAY field before data will be acquired on the channels specified in the CHANNEL LIST field.

9) The "snap-in" real-time digital filtering, stimulus & response, and "monitor mode" (input loopbacked to output) functions are disabled—these functions are only enabled for hardware with an onboard DSP chip.

3.2 Ariel

Ariel hardware can be basically subdivided into two categories: high-end boards with powerful dual-channel analog I/O capability and large amounts of DSP memory, and low-end boards with limited analog I/O and DSP memory. Boards in the latter category usually retail for significantly less than $1000 and are designated by "PC-xxx", where "xxx" refers in some way to the DSP chip on the board. In general, Ariel hardware is known for excellent analog design, including low noise floors and low total harmonic distortion.

Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is also the Ariel factory default for the DSP-16+ and PC-C25 boards. However, other Ariel boards come with factory default I/O base addresses of 300H and 340H, requiring

you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

When multiple Ariel boards are to be used in a system (i.e. when separate boards are dedicated to separate real-time functions) the I/O Base Address used by the software expects the system boards to be at base address plus multiples of 10H. For example, if board 1 is at 33CH, then board 2 would be at 34CH (and board 3 at 35CH, etc.). Note that the system has no way of knowing if two boards from the same manufacturer are actually present in the system unless they are both selected at least once in the ANALOG CONVERSION, DSP ACCELERATOR, DIGITAL SCOPE, and SPECTRUM ANALYZER hardware fields. If you have jumpered two boards at base address and base address + 10H, and then mention only the second board in the above functions, the program will incorrectly attempt to locate this board at the base address. In this discussion, “board 1" is the first board to appear on the menu (from the top down), ”board 2" the next, etc.

3.2.1 Ariel DSP-16+ (TMS320C25)

Basic Descriptive and Installation Details

Use a hardware designator of ‘ARI320’ in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The Ariel DSP-16+ board contains two 16-bit input and two 16-bit output channels and a TMS320C25 DSP chip operating at 40 MHz. It is a full-length board with 8-bit ISA bus XT-style connector.

The two 16-bit input channels on the Ariel DSP-16+ can sample at up to 50 kHz each; if a DSP-16+ board with the 64K memory upgrade (see Onboard Memory Usage discussion below) is being used with a reasonably fast system (25 MHz with 28 msec. avg. access hard disk) stereo disk transfer at 50 kHz should be possible. The board has 3rd-order Chebyshev lowpass anti-aliasing filters on each input channel that roll off relatively slowly after 20 kHz which, while not sufficient in many applications, are suitable in many audio applications where the microphone input provides additional, natural aliasing protection above 20 kHz.

The two 16-bit output channels on the DSP-16+ are capable of over 100 kHz each, although actual hard disk D/A transfer rates are less (see specifications below). Built-in 5-pole Chebyshev low-pass filters with 0.25 dB ripple that roll off at 20 kHz and 2-pole sinx/x reconstruction filters assure very-high-quality output frequency content.

Initial default Hypersignal memory base address for the Ariel DSP-16+ board is D000H and initial I/O base address is 33CH. Usually a memory segment base address of D000H is recommended for IBM-AT and compatible machines, and an address of E000H for IBM-PC and compatible machines. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS and MEM BASE ADDRESS fields, for details on changing these addresses.

NOTE: In many late-model systems, part or all of the D000H and E000H 64 kbyte memory segments are needed for a network card, “SuperVGA” video card, and/or EMS memory. We suggest using the E000H segment for EMS memory or the network card, and placing the video card in a mode that doesn’t require extra memory, leaving the D000H 64 kbyte segment completely free for use by the DSP-16+. Unfortunately, if you are using the DSP-16+ with its full memory-complement (see Onboard Memory Usage, below), then it is probably not possible to enable operation of EMS memory and a network card at the same time.

Also, be sure that you “block out” the region from D000-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the DSP-16+.

Other non-Ariel boards that expect to operate in the same memory segment (either D000H) or E000H) as the DSP-16+ board may coexist in the system *only if* they have a memory enable/disable feature. The DSP-16+ has this feature, which allows enabling of program/data memory prior to access and disabling immediately afterwards. Note that this implies that the memory enable/disable must be controlled through the board’s *I/O address*, which must be unique for each board. This also implies that the two boards cannot perform their intended function at the same time. For example, if the other board was a network card, then you could not make any network accesses when using Hypersignal. Finally, boards with this feature are quite rare.

External triggering is supported by the Analog Conversion signal acquisition / generation function, as is two-channel triggering combinations (e.g. using one channel solely for triggering, and the other for data). External triggering is accomplished via an external TTL-level input that can be used to enable the onboard sample rate timer, thus allowing external control over both input and output operation. Entry of 'EXT' in the Analog Conversion TRIGGER MODE, LEVEL field will enable the external TTL-level trigger; specifying a data level threshold can be accomplished with an 'INT' entry (as in 'INT,350'), acquiring data after a certain positive or negative threshold has been reached *(input only);* in this case, all triggering is being done in onboard TMS320C25 code.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C25 code.

NOTE: For Rev. E DSP-16 boards, you must cut pin 5 trace of U2 and tie pin 5 of U78 to ground. This will enable analog inputs when used with software intended for Rev. G boards (start of "DSP-16+" model), such as Hypersignal and current Ariel driver functions.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C25, 100 nsec
Data Format, Wordlength:	fixed-point, 16-bit
2 channel Hard Disk Record:	50 kHz
2 channel Hard Disk Playback:	50 kHz
Max. 2-chn. Spectrum Analyzer Rate:	50 kHz
1024 pt FFT Accel. Disk File Rate:	10.5 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 16D; 8k x 16P
Maximum Ext. Mem. Config:	32k x 16D; 32k x 16P
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	.005%
Voltage Range:	±1V to ±10V; TA
Input Impedance:	UA
Anti-Aliasing Filters:	yes, FC20, Cheby3
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	50 kHz
Min. Sampling Rate:	76.3 Hz
External TTL-Level Trigger:	yes
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	95 dB
THD:	.033%
Voltage Range:	±10V
Output Impedance:	UA
Reconstruction Filters:	yes, FC20, Cheby5
Sin(x)/x Filters:	yes
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	100 kHz
Min. Sampling Rate:	76.3 Hz
External TTL-Level Trigger:	yes

Onboard Memory Usage

The DSP-16+ is shipped standard with 8k x 16 of data memory, 8k x 16 of program memory, and 2 Mbyte of dynamic memory. Hypersignal uses the full 8K of program memory for executable code. No use is made of the dynamic memory as it requires valuable 'C25 time for constant refresh and is difficult to access from the host PC's point of view. The "64K" memory upgrade is highly recommended for the DSP-16+ board, as in this case the extra 24k x 16 of data memory is used for increased buffer sizes and file-based acceleration areas, and the extra 24k x 16 of program memory is used for convolution acceleration "h[n]" waveform storage and stimulus and response stimuli storage. With the 64K memory upgrade, continuous transfer rates to/from disk are greatly increased, maximum 2-channel Spectrum Analyzer FFT size is increased, and the stimulus & response and convolution/correlation functions are enabled.

Additional Notes About Hypersignal Software Operation

None.

3.2.2 Ariel PC-C25 (TMS320C25)

Basic Descriptive and Installation Details

Use a hardware designator of 'ARI320-B' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

Although in many ways similar to the DSP-16+ board, above, the PC-C25's analog circuitry is drastically different, relying on a Texas Instruments TLC3204x AIC (analog interface chip) connected to the TMS320C25 serial port to provide a single channel of 14-bit A/D and D/A. It is a 1/2 length board with 8-bit ISA bus XT-style connector.

The TLC3204x device allows the sampling rate to be programmed, 3 levels of input gain to be selected, and contains anti-aliasing and reconstruction bandpass filters that track with the programmed sampling rate. The sampling rate and gain are controllable from Hypersignal menus. One should be aware that programming of the sampling rate is not exact, because the TLC3204x device uses two 6-bit integer counters as "divisors" into a master clock frequency, which allows only a discrete number of possibilities. Normally, input levels can be selected to be ±1.5V, ±3V, and ±6V. Also, the highpass portion of the bandpass filters can be selectively disabled (although this must be done through the Hypersignal configuration file as opposed to data-entry menus). For TLC32044 devices, a sin(x)/x D/A output filter is also incorporated.

Memory and I/O base addressing information is identical to the DSP-16+ board, above, but will be repeated here for convenience. Initial default Hypersignal memory base address for the PC-C25 board is D000H and initial I/O base address is 33CH. Usually a memory segment base address of D000H is recommended for IBM-AT and compatible machines, and an address of E000H for IBM-PC and compatible machines. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS and MEM BASE ADDRESS fields, for details on changing these addresses.

NOTE: In many late-model systems, part or all of the D000H and E000H 64 kbyte memory segments are needed for a network card, "SuperVGA" video card, and/or EMS memory. We suggest using the E000H segment for EMS memory or the network card, and placing the video card in a mode that doesn't require extra memory, leaving the D000H 64 kbyte segment completely free for use by the PC-C25. Unfortunately, if you are using the PC-C25 with its full memory-complement (see Onboard Memory Usage, below), then it is probably not possible to enable operation of EMS memory and a network card at the same time.

Also, be sure that you "block out" the region from D000-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the PC-C25.

Other non-Ariel boards that expect to operate in the same memory segment (either D000H) or E000H) as the PC-C25 board may coexist in the system *only if* they have a memory enable/disable feature. The PC-C25 has this feature, which allows enabling of program/data memory prior to access and disabling immediately afterwards. Note that this implies that the memory enable/disable must be controlled through the board's *I/O address*, which must be unique for each board. This also implies that the two boards cannot perform their intended function at the same time. For example, if the other board was a network card, then you could not make any network accesses when using Hypersignal. Finally, boards with this feature are quite rare.

Use of either the TLC32040 or TLC32044 AIC raises a few items of note that you should be aware of. We have noticed that the AIC chips do not correctly scale input voltage as described in their Texas Instruments data sheets when set to differential input, as is the case with the PC-C25. We have currently set the Hypersignal software to the gain setting that is closest to being correct: ±6V, differential input. Also, be aware that input voltages above around ±4V to the AIC chip may begin to encounter distortion (non-linearity). This problem may be caused by the fact that the TLC3204x on these boards is running on ±5V power supply levels, which might allow linear operation only up to some percentage of these levels.

To summarize, an INPUT RANGE field setting of 6 in the Hypersignal Digital Oscilloscope and Spectrum Analyzer fields will allow reasonably accurate measurement of input voltages, but keep your input below ±4V to avoid distortion.

If you change the Digital Oscilloscope or Spectrum Analyzer INPUT RANGE fields to 3, you will increase the gain of the AIC chip to 2 (input now at ±3V full-scale) when using those functions.

If you must change bits in the AIC control word directly, in order to affect AIC settings such as filter enable/disables, loopback, etc, you can use a text-editor to change the "AIC CONTROL WORD" inside the "HSMACRO.CNF" (or "HSACOUS.CNF") ASCII configuration file. See the file "CONFIG.DOC" for a list of configuration file parameters and also an example of how to access this file via external program. See Texas Instruments documentation for details on the available control bits in the AIC control word.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -8191 to 8191 for 14-bit A/D) and applied digitally, via TMS320C25 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C25, 100 nsec
Data Format, Wordlength:	fixed-point, 16-bit
1 channel Hard Disk Record:	19.2 kHz
1 channel Hard Disk Playback:	19.2 kHz
Max. 2-chn. Spectrum Analyzer Rate:	19.2 kHz
1024 pt FFT Accel. Disk File Rate:	10.8 frames/sec
Spectrum Analyzer Noise Floor:	67 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 16 D; 8k x 16 P
Maximum Ext. Mem. Config:	32k x 16 D; 32k x 16 P
Ext. Sync Serial Interface:	yes (used by AIC)
Ext Async. Serial Interface:	no
External Parallel Interface:	yes
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x (AIC)
Codec channels:	none
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±1.5V to ±6V; P3
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby7, JB (AIC)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x (AIC)
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±6V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby7, JB (AIC)
Sin(x)/x Filters:	yes, AIC (TLC32044)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The PC-C25 is shipped standard with 8k x 16 of data memory and 8k x 16 of program memory. Hypersignal uses the full 8K of program memory for executable code. The “64K” memory upgrade is highly recommended for the PC-C25 board, as in this case the extra 24k x 16 of data memory is used for increased buffer sizes and file-based acceleration areas, and the extra 24k x 16 of program memory is used for convolution/correlation acceleration “h[n]” waveform storage and stimulus and response stimuli storage. Also, stimulus & response and convolution/correlation functions are enabled, and speed of data transfer to/from disk is improved, although this has minimal impact since the AIC has a limited sampling rate in the first place.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the PC-C25 board in the Analog Conversion function (i.e. entry of ‘EXT’ for TRIGGER MODE, LEVEL field is ignored).

3.2.3 Ariel DSP-56 (DSP56001)

Basic Descriptive and Installation Details

Use a hardware designator of ‘ARI56-A’ (or just ‘ARI56’) in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The DSP-56 contains the 24-bit fixed-point Motorola DSP chip, along with two 16-bit analog I/O channels. It is a full-length board with 8-bit ISA bus XT-style connector.

Analog input uses sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rates, and linear phase. Analog output provides programmable sampling rates,

9th-order elliptic reconstruction filters (fixed 20 kHz cutoff-frequency), and sin(x)/x filters (fixed 48 kHz cutoff-frequency). The programmable sampling rates on this board must be one of 12 discrete values in a range from 2 to 100 kHz allowed by combinations of three onboard crystal oscillators. Values such as 8, 50, and 100 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports 12-bit 128, 200, and 400 kHz sample modes (although data continuity in any disk-based operations is unlikely at these rates).

NOTE: Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the DSP-56 board comes with a factory default I/O base address of 340H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock or non-standard crystal oscillators and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage. If you change oscillator values, then you will need to make minor modifications to the DSP56 source code in order to program the sampling rate latch correctly (because the built-in Hypersignal programming is disabled by the '-EXT' selection). See section 4.3, Motorola DSP56001 DSP Source Code Interface.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP56001 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP56001, 100 nsec
Data Format, Wordlength:	fixed-point, 24-bit
1 channel Hard Disk Record:	100 kHz
1 channel Hard Disk Playback:	90 kHz
Max. 2-chn. Spectrum Analyzer Rate:	50 kHz
1024 pt FFT Accel. Disk File Rate:	14.8 frames/sec
Spectrum Analyzer Noise Floor:	92 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 24 P, X, Y
Maximum Ext. Mem. Config:	64k x 24 P, X, Y
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes, DSPnet
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits or 2, 12-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H	yes
Typical Dynamic Range:	92 dB
THD:	UA
Voltage Range:	±1V to ±5V; TA
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P17
External Sample Clock:	yes
Max. Sampling Rate:	100 kHz 16-bit, 400 kHz 12-bit
Min. Sampling Rate:	2 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB
THD:	.033%
Voltage Range:	±5V
Output Impedance:	UA
Reconstruction Filters:	yes, ellip9, FC20, JB
Sin(x)/x Filters:	yes, FC48, JB
Programmable Sampling Rate:	yes, P12
External Sample Clock:	yes
Max. Sampling Rate:	100 kHz
Min. Sampling Rate:	2 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The DSP-56 board is shipped standard with 16k x 24 of program, X: data, and Y: data memory. Hypersignal uses less than 4k x 24 of program memory, and the full X: and Y: data memory for analog buffers. An upgrade to 64k x 24 of X: and Y: data memory greatly improves stimulus & response operation by providing up to 48K of memory for stimulus storage, and also enable convolution/correlation acceleration.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the DSP-56 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

3.2.4 Ariel PC-56 (DSP56001)

Basic Descriptive and Installation Details

Use a hardware designator of 'ARI56-B' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The analog circuitry of the basic PC-56 board relies on a Texas Instruments TLC3204x AIC (analog interface chip) connected to the DSP56001 SSI serial channel to provide a single channel of 14-bit A/D and D/A. It is a 1/2 length board with 8-bit ISA bus XT-style connector.

The TLC3204x device allows the sampling rate to be programmed, 3 levels of input gain to be selected, and contains anti-aliasing and reconstruction bandpass filters that track with the programmed sampling rate. The sampling rate and gain are controllable from Hypersignal menus. One should be aware that programming of the sampling rate is not exact, because the TLC3204x device uses two 6-bit integer counters as "divisors" into a master clock frequency, which allows only a discrete number of possibilities. Normally, input levels can be selected to be ±1.5V, ±3V, and ±6V. Also, the highpass portion of the bandpass filters can be selectively disabled (although this must be done through the Hypersignal configuration file as opposed to data-entry menus). For TLC32044 devices, a sin(x)/x D/A output filter is also incorporated.

NOTE: Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the PC-56 board comes with a factory default I/O base address of 340H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

Use of either the TLC32040 or TLC32044 AIC raises a few items of note that you should be aware of. We have noticed that the AIC chips do not correctly scale input voltage as described in their Texas Instruments data sheets when set to differential input mode, as is the case with the PC-56. We have currently set the Hypersignal software to the gain setting that is closest to being correct: ±6V, differential input. Also, be aware that input voltages above around ±4V to the AIC chip may begin to encounter distortion (non-linearity). This problem may be caused by the fact that the TLC3204x on these boards is running on ±5V power supply levels, which might allow linear operation only up to some percentage of these levels.

To summarize, an INPUT RANGE field setting of 6 in the Hypersignal Digital Oscilloscope and Spectrum Analyzer fields will allow reasonalby accurate measurement of input voltages, but keep your input below ±4V to avoid distortion.

If you change the Digital Oscilloscope or Spectrum Analyzer INPUT RANGE field to 3, you will increase the gain of the AIC chip to 2 (input now at ±3V full-scale).

If you must change bits in the AIC control word directly, in order to affect AIC settings such as filter enable/disables, loopback, etc. you can use a text-editor to change the "AIC CONTROL WORD" inside the "HSMACRO.CNF" (or "HSACOUS.CNF") ASCII configuration file. See the file "CONFIG.DOC" for a list of configuration file parameters and also an example of how to access this file via external program. See Texas Instruments documentation for details on the available control bits in the AIC control word.

Also, on the Ariel PC-56 board, there can be a problem with Rev. D and later boards that affects amplitude read-out. Because Ariel started using TLC32044s instead of the older TLC32041 (due to an anti-alias filter problem with the chip), newer PC-56 boards are not supposed to have the "Z1" zener-diode and "R20" resistor installed. However, it appears that some new boards, with TLC32044s, still have the R20 resistor installed by mistake. You should check your board to see if this is the case. For information, the anti-alias filter problem is that TLC32041 AICs don't disable the highpass portion of the bandpass filter when programmed to do so by the chip's "control word".

To summarize, if your PC-56 has a TLC32041 AIC chip, then Z1 and R20 should be installed. If your PC-56 has a TLC32044 (or TLC32046), then Z1 and R20 should be removed.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -8191 to 8191 for 14-bit A/D) and applied digitally, via DSP56001 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP56001, 100 nsec
Data Format, Wordlength:	fixed-point, 24-bit
1 channel Hard Disk Record:	19.2 kHz
1 channel Hard Disk Playback:	19.2 kHz
Max. 2-chn. Spectrum Analyzer Rate:	19.2 kHz
1024 pt FFT Accel. Disk File Rate:	14.8 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 24 P, 4k x 24 X,Y
Maximum Ext. Mem. Config:	32k x 24 P, 16k x 24 X,Y
Ext. Sync Serial Interface:	yes (used by AIC)
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x (AIC)
Codec channels:	none
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±1.5V to ±6V; P3
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby7, JB (AIC)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x (AIC)
Codec channels:	none
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±6V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby7, JB (AIC)
Sin(x)/x Filters:	yes, AIC (TLC32044)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The PC-56 board is shipped standard with 8k x 24 of program and 4k x 24 of X: data and Y: data memory. Hypersignal uses less than 4k x 24 of program memory, and the full X: and Y: data memory for analog buffers. An X: and Y: data memory upgrade would greatly improve all phases of Hypersignal operation, especially maximum FFT size (both file-based acceleration and Spectrum Analyzer) and stimulus & response operation. Speed of data transfers to/from disk is also improved, but this has minimal impact since the AIC has a limited sampling rate in the first place. Also, the memory upgrade will enable acceleration of Convolution/Correlation operations.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the PC-56 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

3.2.5 Ariel PC-56D (DSP56001)

Basic Descriptive and Installation Details

Use a hardware designator of 'ARI56-D' in the Hypersignal software System Configuration menu for use with Ariel ProPort or DATPort systems, 'ARI56-DM' for use with Ariel Digital Microphone, and 'ARI56-B' for use with onboard analog I/O (see notes below). It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled. The PC-56D and ProPort, DATPort, or Digital Microphone combinations are only active with Hypersignal-Acoustic software.

The PC-56D contains the 24-bit fixed-point Motorola DSP56001 DSP chip, along with onboard analog I/O in the form of either a TLC3204x AIC (analog interface chip) connected to the DSP56001 SSI serial channel, which provides one channel of 14-bit A/D and D/A, "ProPort" external analog I/O unit, "DATPort" AES/EBU external unit, or Digital Microphone. The latter three options are also connected to the SSI serial channel, using the external "DSP Port" connector. Note that all four analog I/O options are mutually exclusive, since they each require the one available SSI channel to operate. The PC-56D is a 1/2 length board with 8-bit ISA bus XT-style connector.

The TLC3204x AIC analog I/O has exactly the same specifications as the standard PC-56 board, and is described fully in section 3.2.4 above, Ariel PC-56.

ProPort analog input is based on sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rates, and linear phase. Input rates are software-selectable from 8 kHz to 96 kHz at 16-bits on both channels. In addition, a backpanel switch can be set to emable an input rate of 384 kHz, but at only 12-bits and one channel. Analog output provides programmable sampling rates, and 8x oversampling, with software-selectable rates from 8 kHz to 48 kHz. The oversampling means that, even at the minimum output rate of 8 kHz, the actual output sampling rate will always be greater than 48 kHz, so Ariel has installed high-order reconstruction and sin(x)/x output filters fixed at 48 kHz. Additionally, as part of its design intended for acoustic and audio professionals, the ProPort offers connectors for both line-level and microphone input, dual potentiometer-adjustable gain controls for microphone the inputs, and switchable "phantom power" for the microphone inputs.

Hypersignal allows you to use any of the available analog I/O options with the PC-56D. Enter 'ARI56-B' in the System Configuration menu to use the AIC analog I/O, enter 'ARI56-D' to use the ProPort or DATPort options, and enter 'ARI56-DM' for the Digital Microphone. Depending on the System Configuration menu entry(s), Hypersignal programs the PC-56D board to con-

nect the SSI serial channel to either the onboard analog I/O (TLC32044 AIC) or external serial port connector (DSP Port). When using the onboard analog I/O, you should also enter the clock frequency of the board (probably 27.648 MHz) in the System Configuration menu. This is done by adding it after the board entry, for example, enter 'ARI56-B-27.648'. Reason: because otherwise the software assumes a clock rate for the standard PC-56 board (probably 20.7 MHz; see section 3.2.4 above), causing software time and frequency axis readouts to be incorrect. See the online Help for System Configuration for details on entering a clock frequency.

PROPORT NOTES: Ariel designed the ProPort so it could be used with other DSP boards besides the PC-56D. Due to this flexibility, and its modular construction, there are three very important items that you must be aware of in order to achieve correct ProPort operation with the PC-56D, and indeed, to avoid damage to your system:

1) The Data Format switch on the back of the ProPort should be set to "A".

2) You must ensure that your ProPort serial-port cable is really a PC-56D cable. If you use a cable designed for a different board, you can damage your entire system, because other cables will connect +5V on the PC-56D directly to ground. Both ends of the PC-56D cable are identical, and can be plugged into either the PC-56D or the ProPort.

3) VERY IMPORTANT: We have found that ProPort inputs are *extremely sensitive* to over-voltage, and can be easily blown. We strongly recommend that you:

a. Do not turn off the ProPort with input signal(s) still applied—turn off the signal source first! You should probably go so far as to unplug any signal sources.

b. Be very, very careful with static electricity; especially when touching the ProPort. Because of the non-grounded AC power supply system used for the ProPort, it cannot discharge static electricity well. We suggest touching the metal frame of your compurter or other well-grounded metal surface before you touch the ProPort.

The programmable sampling rates for 16-bit analog I/O with the ProPort must be one of 7 discrete values in a range from 8 to 96 kHz. Values such as 8, 16, and 48 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports a 12-bit, one-channel 384 kHz sampling rate mode when the ProPort backpanel switch is set to position 9 (although data continuity in any disk-based operations is unlikely at this rate). A System Configuration menu selection of 'ARI56-DX' assumes the topmost two programmable rates are 100 kHz and 50 kHz, instead of 96 kHz and 48 kHz; this selection is only accurate if the 24.576 MHz oscillator (one of three different values) inside the ProPort has been changed to 25.6 MHz. Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP56001 Source Code Interface (see section 4.3, Motorola DSP5600x DSP Source Code Interface), it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion. Also, there are three oscillators inside the ProPort that are used to provide the different sampling rates. One or more of these values can be replaced to achieve exact, desired real-time sampling rates with optimum performance (no extra signal processing overhead).

When used with the DATPort interface, Hypersignal does not actually set (via the PC-56D) the sampling rate and other controls, which must instead be set directly on the DATPort unit. However, the sampling rate entered in Hypersignal menus should match the DATPort unit setting, so that units readout will be correct. Likewise, input gain and other settings cannot be controlled from Hypersignal menus, but must be set on the DATPort.

When used with the Digital Microphone, different sampling rates are possible than with the ProPort. Consult the Digital Microphone documentation for details. Hypersignal "knows" these rates, and programs the PC-56D accordingly when a 'ARI56-DM' entry is used in the System Configuration menu. Note also that the Digital Microphone offers dual auxiliary inputs, in the form of two mini-stereo jacks, that can be used for non-microphone line-level input.

NOTE: Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the PC-56D board comes with a factory default I/O base address of 340H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP56001 code.

Hardware Ratings and Software Throughput Specifications

The analog specifications listed below pertain to the ProPort and Digital Microphone, and are separated by a '/' when different. For analog specifications when the TLC3204x analog I/O option on the PC-56D board is selected, see section 3.2.4 above, Ariel PC-56.

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP56001, 75 nsec
Data Format, Wordlength:	fixed-point, 24-bit
1 channel Hard Disk Record:	96 kHz / 88.2 kHz
1 channel Hard Disk Playback:	48 kHz / NA
Max. 2-chn. Spectrum Analyzer Rate:	96 kHz / 88.2 kHz
1024 pt FFT Accel. Disk File Rate:	14.8 frames/sec
Spectrum Analyzer Noise Floor:	90 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 24 P, 4k x 24 X,Y
Maximum Ext. Mem. Config:	32k x 24 P, 16k x 24 X,Y
Ext. Sync Serial Interface:	yes ("DSP Port" conn. if no analog I/O option)
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes (DSPNet)
Host Data Transfer Method:	Prog. I/O Mapped
Number of linear PCM channels:	2, 16-bits (96 kHz) or 1, 12-bits (384 kHz) / 2, 16-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	.005% / .005% (1 kHz)
Voltage Range:	±1V; PA (microphone), ±3V; (line-level) / ±3V; (aux input)
Input Impedance:	2.2 kOhm, 5 kOhm (aux input)
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P7 / yes, P8
External Sample Clock:	yes
Max. Sampling Rate:	96 kHz 16-bit, 384 kHz 12-bit / 88.2 kHz
Min. Sampling Rate:	8 kHz / 689 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits / NA
Codec channels:	none / NA
Expandable:	yes, DSPnet
Simultaneous S/H:	yes / NA
Typical Dynamic Range:	90 dB / NA
THD:	0.033% (10 kHz) / NA
Voltage Range:	±3V
Output Impedance:	300 Ohm / NA
Reconstruction Filters:	yes, ellip9, FC48 / NA
Sin(x)/x Filters:	yes, FC48 / NA
Programmable Sampling Rate:	yes, P7 / NA
External Sample Clock:	yes / NA
Max. Sampling Rate:	48 kHz 16-bit / NA
Min. Sampling Rate:	8 kHz / NA
External TTL-Level Trigger:	no / NA

Onboard Memory Usage

The PC-56D board is shipped standard with 8k x 24 of program and 4k x 24 of X: data and Y: data memory. Hypersignal uses less than 4k x 24 of program memory when the TLC3204x analog option has been selected, less than 5k x 24 of program memory when the ProPort or Digital Microphone has been selected, and the full X: and Y: data memory for analog buffers. An X: and Y: data memory upgrade is highly recommended, as it greatly improves all phases of Hypersignal operation, especially disk transfer at high sampling rates (40 kHz and above), maximum FFT size (both file-based acceleration and Spectrum Analyzer), and stimulus & response operation. Also, the memory upgrade will enable accelerated convolution/correlation operations. Speed of data transfers to/from disk can be vastly improved, which can be very important when the PC-56D is being used with the ProPort or DATPort interfaces. Depending on the speed of the host system, a memory upgrade may have minimal impact when the AIC analog I/O option is in use, since the AIC has a limited sampling rate in the first place.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the PC-56 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

3.2.6 Ariel DSP-32C (DSP32C)

Basic Descriptive and Installation Details

Use a hardware designator of 'ARI32C' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The DSP-32C contains the 32-bit floating-point AT&T DSP32C DSP chip, along with two 16-bit analog I/O channels. It is a full-length board with 16-bit ISA bus AT-style connector.

Analog input uses sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rates, and linear phase. Analog output provides programmable sampling rates, 9th-order elliptic reconstruction filters (fixed 20 kHz cutoff-frequency), and sin(x)/x filters (fixed 48 kHz cutoff-frequency). The programmable sampling rates on this board must be one of 12 discrete values in a range from 2 to 100 kHz allowed by combinations of three onboard crystal oscillators. Values such as 8, 50, and 100 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports 12-bit 128, 200, and 400 kHz sample modes (although data continuity in any disk-based operations is unlikely at these rates).

NOTE: Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the DSP-32C board comes with a factory default I/O base address of 300H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

For proper operation of the Ariel DSP-32C board, please be aware that:

1) The DSP-32C must be installed in an AT-style bus machine due to the 16-bit host interface used.

2) On some machines, problems with the DSP-32C and Hypersignal have occurred when running with a high bus speed (above 8 MHz). The symptoms of this are machine lock-up while running a Hypersignal real-time function or accelerated DSP function, an error message of the form "...mem < 1024 K..." and/or grossly incorrect data displays. The first correction to try is disabling the 80286 INSTRUCTION SET field in the System Configuration menu. The second correction is to slow the bus speed of your machine; some machines have a switch on the motherboard which will allow you to choose bus speed settings. Consult the documentation for your machine to see if this is the case.

3) Earlier revision DSP-32Cs had a host-bus interface problem that could cause interference with video cards using D000H memory, and sometimes could prevent the host machine from booting altogether. If you suspect a general problem such as this please contact Hyperception, Signalogic, or Ariel (be ready to provide the board revision number).

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock or non-standard crystal oscillators and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage. If you change oscillator values, then you will need to make minor modifica-

tions to the DSP32C source code in order to program the sampling rate latch correctly (because the built-in Hypersignal programming is disabled by the '-EXT' selection). See section 4.4, AT&T WE DSP32C DSP Source Code Interface.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32C code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	100 kHz
1 channel Hard Disk Playback:	100 kHz
Max. 2-chn. Spectrum Analyzer Rate:	100 kHz
1024 pt FFT Accel. Disk File Rate:	19 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	64k x 32
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPnet
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits or 2, 12-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	UA
Voltage Range:	±1V to ±5V; TA
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P17
External Sample Clock:	yes
Max. Sampling Rate:	100 kHz 16-bit, 400 kHz 12-bit
Min. Sampling Rate:	2 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB
THD:	.033%
Voltage Range:	±5V
Output Impedance:	UA
Reconstruction Filters:	yes, ellip9, FC20, JB
Sin(x)/x Filters:	yes, FC48, JB
Programmable Sampling Rate:	yes, P12
External Sample Clock:	yes
Max. Sampling Rate:	100 kHz 16-bit
Min. Sampling Rate:	2 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The basic DSP-32C board contains 16k x 32 of zero wait-state external A area memory. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog data buffers and file-based acceleration processing. The DSP-32C memory can be expanded by changing memory chips to 64k x 32 of total external A memory. We strongly suggest the 64k x 32 amount in order to enable all Hypersignal acceleration functions and allow continuous real-time functions to operate at full speed and capability. For example, 12k x 32 of data memory is not enough to support 2-channel 4096 pt. FFT Spectrum Analyzer operation, but will allow smaller FFT sizes and/or 1-channel operation. Also, 64k x 32 of total memory will enable convolution/correlation acceleration, and greatly increase maximum output stimulus sizes for stimulus & response operation.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the DSP-32C board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

3.2.7 Ariel PC-32C (DSP32C)

Basic Descriptive and Installation Details

Use a hardware designator of 'ARI32C-B' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine. The PC-32C system option is only active with Hypersignal-Acoustic software.

The PC-32C contains the 32-bit floating-point AT&T DSP32C DSP chip, and no onboard analog I/O—instead, Hypersignal supports this board when used with the Ariel "DATPort" digital interface or the "ProPort" external analog I/O unit. The PC-32C is a 1/2 length board with 8-bit ISA bus XT-style connector.

ProPort analog input is based on sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rates, and linear phase. Input rates are switch-selectable from 8 kHz to 96 kHz at 16-bits on both channels. In addition, a backpanel switch can be set to emable an input rate of 384 kHz, but at only 12-bits and one channel. Analog output provides programmable sampling rates, and 8x oversampling, with switch-selectable rates from 8 kHz to 48 kHz. The oversampling means that, even at the minimum output rate of 8 kHz, the actual output sampling rate will always be greater than 48 kHz, so Ariel has installed high-order reconstruction and sin(x)/x output filters fixed at 48 kHz. Additionally, as part of its design intended for acoustic and audio professionals, the ProPort offers connectors for both line-level and microphone input, dual potentiometer-adjustable gain controls for microphone the inputs, and switchable "phantom power" for the microphone inputs.

Sampling rates for the ProPort when used with the PC-32C board are not software-selectable. You must select the rate using a rotary switch located on the ProPort backpanel, and then ensure that matching values are entered into SAMPLING FREQUENCY fields in the Hypersignal software. Otherwise, time and frequency axis units and readouts may not be correct. See below for detailed information about available sampling rates.

When used with the DATPort interface, Hypersignal does not actually set (via the PC-32C) the sampling rate and other controls, which must instead be set directly on the DATPort unit. However, the sampling rate entered in Hypersignal menus should match the setting DATPort unit setting, so that units readout will be correct. Likewise, input gain and other settings cannot be controlled from Hypersignal menus, but must be set on the DATPort.

PROPORT NOTES: **Ariel designed the ProPort so it could be used with other DSP boards besides the PC-32C. Due to this flexibility, and its modular construction, there are three very important items that you must be aware of in order to achieve correct ProPort operation with the PC-32C, and indeed, to avoid damage to your system:**

1) The Data Format switch on the back of the ProPort should be set to "B".

2) You must ensure that your ProPort serial-port cable is really a PC-32C cable. If you use a cable designed for a different board, you can damage your entire system, because other cables will connect +5V on the PC-32C directly to ground. Note also that the ends of the cable are not identical; each end should be clearly marked as either PC-32C or ProPort.

3) VERY IMPORTANT: We have found that ProPort inputs are *extremely sensitive* to over-voltage, and can be easily blown. We strongly recommend that you:

a. Do not turn off the ProPort with input signal(s) still applied—turn off the signal source first! You should probably go so far as to unplug the signal source.

b. Be very, very careful with static electricity; especially when touching the ProPort. Because of the non-grounded AC power supply system used for the ProPort, it cannot discharge static electricity well. We suggest touching the metal frame of your compurter or other well-grounded metal surface before you touch the ProPort.

The backpanel switch-selectable sampling rates for 16-bit analog I/O operation with the ProPort must be one of 7 discrete values in a range from 8 to 96 kHz. Values such as 8, 16, and 48 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports a 12-bit, one-channel 384 kHz sampling rate mode when the ProPort backpanel switch is set to position 9 (although data continuity in any disk-based operations is unlikely at these rates). Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP32C Source Code Interface (see section 4.4, AT&T WE DSP32C DSP Source Code Interface), it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion. Also, there are three oscillators inside the ProPort that are used to provide the different sampling rates. One or more of these values can be replaced with to achieve exact, desired real-time sampling rates with optimum performance (no extra signal processing overhead).

NOTE: **Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the PC-32C board comes with a factory default I/O base address of 300H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.**

For proper operation of the Ariel PC-32C board, please be aware that it must be installed in an AT-style bus machine due to the 16-bit host interface connector used. It is also necessary to set the 80286 INSTRUCTION SET field in this menu to 'Y' (which meansu you must have a 80286, 80386, or 80486 machine).

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time and frequency units display and readout.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32C code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	96 kHz
1 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	96 kHz
1024 pt FFT Accel. Disk File Rate:	19 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	64k x 32
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPnet
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits (96 kHz) or 1, 12-bits (384 kHz)
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	UA
Voltage Range:	±1V; PA (microphone), ±3V; (line-level)
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	no, SWS7
External Sample Clock:	yes
Max. Sampling Rate:	96 kHz 16-bit, 384 kHz 12-bit
Min. Sampling Rate:	8 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, ellip9, FC48
Sin(x)/x Filters:	yes, FC48
Programmable Sampling Rate:	no, SWS7
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz 16-bit
Min. Sampling Rate:	8 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The basic PC-32C board contains 16k x 32 of zero wait-state external A area memory. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog data buffers and file-based acceleration processing. The PC-32C memory can be expanded by changing memory chips to 64k x 32 of total external A memory. We strongly suggest the 64k x 32 amount in order to enable all Hypersignal acceleration functions and allow continuous real-time functions to operate at full speed and capability. For example, 12k x 32 of data memory is not enough to support 2-channel 4096 pt. FFT Spectrum Analyzer operation, but will allow smaller FFT sizes and/or 1-channel operation. Also, 64k x 32 of total memory will enable convolution/correlation acceleration, and greatly increase maximum output stimulus sizes for stimulus & response operation.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the PC-32C board in the Analog Conversion function (i.e. entry of ‘EXT’ for TRIGGER MODE, LEVEL field is ignored).

3.2.8 Ariel DSP-96 (DSP96002)

Basic Descriptive and Installation Details

Use a hardware designator of 'ARI96' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The DSP-96 contains the 32-bit floating-point Motorola DSP96002 DSP chip, along with two 16-bit analog I/O channels. It is a full-length board with 16-bit ISA bus AT-style connector.

Analog input uses sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rate, and linear phase. Analog output uses delta-sigma D/A converters, which provide programmable sampling rates, built-in reconstruction filters that track exactly with the programmed sampling rate, and linear phase. Note that the analog I/O for the DSP-96 is not actually contained on the DSP-96 board itself, but on a daughterboard. Programmable *input* sampling rates must be one of 12 discrete values in a range from 6 to 96 kHz allowed by combinations of three onboard crystal oscillators. Values such as 8, 48, and 96 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports one-channel, 12-bit 128, 176.4, 192, 256, and 384 kHz input sample modes (although data continuity in any disk-based operations is unlikely at these rates). Programmable *output* sampling rates must be one of 10 discrete values in a range from 6 to 75 kHz.

NOTE: Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the DSP-96 board comes with a factory default I/O base address of 140H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP96002 code.

Hardware Ratings and Software Throughput Specifications

The specifications below are based on analog I/O daughtercard installed for DSP-96 board.

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP96002, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	96 kHz
1 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	128 kHz
1024 pt FFT Accel. Disk File Rate:	18.5 frames/sec
Spectrum Analyzer Noise Floor:	85 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	256k x 32 DRAM Port A, 16k x 32 SRAM Port A,B
Maximum Ext. Mem. Config:	16M x 32 DRAM Port A, 64k x 32 SRAM Port A,B
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPnet
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits (96 kHz) or 1, 12-bits (384 kHz)
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	0.008%
Voltage Range:	±.5V to ±5.5V; TA
Input Impedance:	14.5 kOhm
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P17
External Sample Clock:	yes
Max. Sampling Rate:	96 kHz 16-bit, 384 kHz 12-bit
Min. Sampling Rate:	6 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB
THD:	0.007%
Voltage Range:	±4V
Output Impedance:	UA
Reconstruction Filters:	yes, DS
Sin(x)/x Filters:	yes, DS
Programmable Sampling Rate:	yes, P10
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz 16-bit
Min. Sampling Rate:	6 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The DSP-96 is shipped standard with a huge amount of memory, and flexible mapping possibilities for using this memory. Since there are so many memory mapping configurations for the DSP-96, it is better to think of its memory as three physical “banks”. These banks can be placed in X:, Y:, and P: memory in different ways, including both “vector” and “scalar” configurations. However, Hypersignal always uses a vector configuration, since this provides for fastest execution of most DSP programs. In this configuration, “bank 1", containing 256k x 32 of dynamic RAM (DRAM) in the standard DSP-96 shipment, serves as external data memory. DRAM contents are automatically maintained by built-in refresh circuitry on the DSP96002 chip. The ”long word" address for bank 1 is used, so that the X: and Y: “halves” of bank 1 have the same address (which means that there is 128k x 32 of both X: and Y: memory starting at 20400000H). Bank 3, containing 16k x 32 of static RAM (SRAM) in the standard DSP-96 shipment, is then used as external program memory (starting at 40000000H). Note that bank 2, containing another 16k x 32 of SRAM in standard shipments, is not used at all by Hypersignal! This configuration offers several advantages:

1) external program memory is in SRAM, which requires no wait-states

2) external data memory is set up in true vector fashion, with X: and Y: areas being independently accessible, and starting at the same base address

3) bank 3, used for external program memory, is accessed through a different physical “port” to the outside world than bank 1 (data memory); since data and program memory accesses are thus on separate buses, the DSP96002 can fetch the next instruction in parallel with data, providing further speed improvements

4) the unused bank 2 remains free for user-defined applications

Hypersignal uses less than 8k x 32 of the 16k x 32 of bank 3 program memory (which can be expanded to 64k x 32). Because the external data memory is so large, all Hypersignal analog I/O and acceleration function operate to their maximum capability in the standard DSP-96 configuration. The only function that could benefit from data memory expansion (which could be up to a maximum of 16M x 32) is the stimulus & response operation, which could output even larger stimuli waveforms.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the DSP-96 board in the Analog Conversion function (i.e. entry of ‘EXT’ for TRIGGER MODE, LEVEL field is ignored).

3.2.9 Ariel MP3210 (DSP3210)

Basic Descriptive and Installation Details

Use a hardware designator of 'ARI3210' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The MP3210 contains one or two 32-bit floating-point AT&T DSP3210 DSP chips, along with two 16-bit analog I/O channels on an analog I/O daughterboard, large amount of DRAM, and various external I/O connectors, including DT-Connect. It is a full-length board with 16-bit ISA bus AT-style connector. The factory configuration of the MP3210 has only one DSP3210 device, but a second one can be installed at any time, along with a PAL change.

Daughterboard analog input uses sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rate, and linear phase. Analog output uses delta-sigma D/A converters, which provide programmable sampling rates, built-in reconstruction filters that track exactly with the programmed sampling rate, and linear phase. Note that the analog I/O for the MP3210 is not actually contained on the MP3210 board itself, but on a daughterboard. Programmable *input* sampling rates must be one of 12 discrete values in a range from 6 to 96 kHz allowed by combinations of three onboard crystal oscillators. Values such as 8, 48, and 96 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports one-channel, 12-bit 128, 176.4, 192, 256, and 384 kHz input sample modes (although data continuity in any disk-based operations is unlikely at these rates). Programmable *output* sampling rates must be one of 10 discrete values in a range from 6 to 48 kHz.

The MP3210 can also be used with the Ariel ProPort, in which case System Config hardware designator values should be set to 'ARI3210-B', and the MP3210 must have a "DSP Port" daughterboard installed, rather than the standard analog I/O daughterboard. ProPort analog input is based on sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rates, and linear phase. Input rates are software-selectable from 8 kHz to 96 kHz at 16-bits on both channels. In addition, a backpanel switch can be set to emable an input rate of 384 kHz, but at only 12-bits and one channel. Analog output provides programmable sampling rates, and 8x oversampling, with software-selectable rates from 8 kHz to 48 kHz. The oversampling means that, even at the minimum output rate of 8 kHz, the actual output sampling rate will always be greater than 48 kHz, so Ariel has installed high-order reconstruction and sin(x)/x output filters fixed at 48 kHz. Additionally, as part of its design intended for acoustic and audio professionals, the ProPort offers connectors for both line-level and microphone input, dual potentiometer-adjustable gain controls for microphone the inputs, and switchable "phantom power" for the microphone inputs.

PROPORT NOTES: Ariel designed the ProPort so it could be used with other DSP boards besides the MP3210. Due to this flexibility, and its modular construction, there are three very important items that you must be aware of in order to achieve correct ProPort operation with the MP3210, and indeed, to avoid damage to your system:

1) The Data Format switch on the back of the ProPort should be set to "A".

2) You must ensure that your ProPort serial-port cable is really a MP3210 cable. If you use a cable designed for a different board (such as PC-32C), you can damage your entire system, because other cables will connect +5V on the MP3210 directly to ground. Both ends of the MP3210 cable are identical, and can be plugged into either the MP3210 or the ProPort.

3) VERY IMPORTANT: We have found that ProPort inputs can be *extremely sensitive* to over-voltage, and can be easily blown. We strongly recommend that you:

a. Do not turn off the ProPort with input signal(s) still applied—turn off the signal source first! You should probably go so far as to unplug any signal sources.

b. Be very, very careful with static electricity; especially when touching the ProPort. Because of the non-grounded AC power supply system used for the ProPort, it cannot discharge static electricity well. We suggest touching the metal frame of your compurter or other well-grounded metal surface before you touch the ProPort.

3.2.9 Ariel MP3210 (DSP3210)

The programmable sampling rates for 16-bit analog I/O with the ProPort must be one of 7 discrete values in a range from 8 to 96 kHz. Values such as 8, 16, and 48 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports a 12-bit, one-channel 384 kHz sampling rate mode when the ProPort backpanel switch is set to position 9 (although data continuity in any disk-based operations is unlikely at this rate). Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP3210 Source Code Interface (see section 4.9, AT&T DSP3210 DSP Source Code Interface), it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion. Also, there are three oscillators inside the ProPort that are used to provide the different sampling rates. One or more of these values can be replaced to achieve exact, desired real-time sampling rates with optimum performance (no extra signal processing overhead).

Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the MP3210 board comes with a factory default I/O base address of 340H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address. While initial default Hypersignal memory base address for the MP3210 board is DF00H, the default setting in the System Config menu MEM BASE ADDRESS field for Ariel boards is D000H. Thus, you should change this value to DF00H immediately, when first initializing Hypersignal. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the MEM BASE ADDRESS field, for details on changing this address. Note also that the memory base address is never actually set via jumpers (links) or switches on the board (as is the I/O base address), but is instead written to registers on the board via high-level Hypersignal software.

NOTES: **We highly suggest you not change the default value of DF00H for memory base address. This value will allow the board to work with network cards and/or "SuperVGA" video cards requiring extra "high memory" and located in the D000:0000H segment. This value also preserves E000H-EFFFH as a possible EMS memory segment.**

Also, be sure that you "block out" the region from DF00-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the MP3210.

Use a hardware designator of 'ARI3210/1' to cause Hypersignal functions to execute on the second DSP3210 (processor 1). Note that analog I/O function will no operate, because the A/D and D/A converters are connected only to processor 0. Entry of 'ARI3210/0' (processor 0) is same as default entry of 'ARI3210'.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP3210 code.

Hardware Ratings and Software Throughput Specifications

The specifications below are based on analog I/O daughtercard installed for MP3210 board.

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP3210, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	96 kHz
1 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	70 kHz
1024 pt FFT Accel. Disk File Rate:	18.5 frames/sec
Spectrum Analyzer Noise Floor:	85 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	256k x 32 DRAM Partition B, 16k x 32 SRAM Partition A
Maximum Ext. Mem. Config:	16M x 32 DRAM Partition B, 512k x 32 SRAM Partition A
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes, DSPnet
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits or 1, 12-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	0.008%
Voltage Range:	±.5V to ±5.5V; TA
Input Impedance:	14.5 kOhm
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P17
External Sample Clock:	yes
Max. Sampling Rate:	96 kHz 16-bit, 384 kHz 12-bit
Min. Sampling Rate:	6 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB
THD:	0.007%
Voltage Range:	±4V
Output Impedance:	UA
Reconstruction Filters:	yes, DS
Sin(x)/x Filters:	yes, DS
Programmable Sampling Rate:	yes, P10
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz 16-bit
Min. Sampling Rate:	6 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The standard MP3210 board contains 16k x 32 of one wait-state SRAM memory, located in partition A, and 256k x 32 of 4-6 wait-state DRAM memory, located in partition B. Hypersignal uses less than 4k x 32 of the SRAM memory for program storage, and the remainder for analog data buffers and file-based acceleration processing. The MP3210 memory can be expanded by changing memory chips to 512k x 32 of total SRAM. We strongly suggest the 512k x 32 SRAM amount in order to enable all Hypersignal acceleration functions and allow continuous real-time functions to operate at full speed and capability. For example, 12k x 32 of data memory is not enough to support 2-channel 4096 pt. FFT Spectrum Analyzer operation, but will allow smaller FFT sizes and/or 1-channel operation. Also, 512k x 32 will greatly increase maximum output stimulus sizes for stimulus & response operation. Currently, for performance reasons, Hypersignal does not use the DRAM memory on the MP3210 board. If you need much larger data processing or analog I/O buffer sizes, then you may want to consider modifying the DSP3210 Source Code Interface initialization routine to use DRAM locations for default buffer base addresses (see section 4.9 AT&T DSP3210 DSP Source Code Interface).

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the MP3210 board in the Analog Conversion function (i.e. entry of ‘EXT’ for TRIGGER MODE, LEVEL field is ignored).

3.2.10 Ariel DSP-C40 (TMS320C40)

Basic Descriptive and Installation Details

Use a hardware designator of ‘ARIC40’ in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

3.2.10 Ariel DSP-C40 (TMS320C40)

The DSP-C40 contains the 32-bit floating-point Texas Instruments TMS320C40 DSP chip, along with two 16-bit analog I/O channels contained on an analog I/O daughterboard, large amount of DRAM, and various external I/O connectors, including DT-Connect and Global Bus+. It is a full-length board with 16-bit ISA bus AT-style connector. Hypersignal also supports the DSP-C40U (enter 'ARIC40-B' in System Configuration menu). The DSP-C40U does not have DRAM, Global Bus+, or analog I/O daughterboard).

Daughterboard analog input uses sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rate, and linear phase. Analog output uses delta-sigma D/A converters, which provide programmable sampling rates, built-in reconstruction filters that track exactly with the programmed sampling rate, and linear phase. Note that the analog I/O for the DSP-C40 is not actually contained on the DSP-C40 board itself, but on a daughterboard. Programmable *input* sampling rates must be one of 12 discrete values in a range from 6 to 96 kHz allowed by combinations of three onboard crystal oscillators. Values such as 8, 48, and 96 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports one-channel, 12-bit 128, 176.4, 192, 256, and 384 kHz input sample modes (although data continuity in any disk-based operations is unlikely at these rates). Programmable *output* sampling rates must be one of 10 discrete values in a range from 6 to 48 kHz.

Initial default Hypersignal I/O base address for all Ariel boards is 33CH, which is the Ariel factory default for some boards. However, the DSP-C40 board comes with a factory default I/O base address of 340H, requiring you to change the Hypersignal default immediately. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C40 code.

Hardware Ratings and Software Throughput Specifications

The specifications below are based on analog I/O daughtercard installed for DSP-C40 board.

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C40, 50 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	96 kHz
1 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	96 kHz
1024 pt FFT Accel. Disk File Rate:	23 frames/sec
Spectrum Analyzer Noise Floor:	85 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	256k x 32 DRAM, 32k x 32 SRAM
Maximum Ext. Mem. Config:	16M x 32 DRAM, 6M x 32 SRAM
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes, DSPnet
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits or 1, 12-bit
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H	yes
Typical Dynamic Range:	85 dB
THD:	0.008%
Voltage Range:	±.5V to ±5.5V; TA
Input Impedance:	14.5 kOhm
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P17
External Sample Clock:	yes
Max. Sampling Rate:	96 kHz 16-bit, 384 kHz 12-bit
Min. Sampling Rate:	6 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPnet
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB
THD:	0.007%
Voltage Range:	±4V
Output Impedance:	UA
Reconstruction Filters:	yes, DS
Sin(x)/x Filters:	yes, DS
Programmable Sampling Rate:	yes, P10
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz 16-bit
Min. Sampling Rate:	6 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The standard DSP-C40 board contains two banks of 16k x 32 of zero wait-state SRAM memory, and 256k x 32 of 4-6 wait-state DRAM memory. Hypersignal uses less than 4k x 32 of the SRAM memory for program storage, and the DRAM for analog data buffers and file-based acceleration processing (unless DSP-C40U selection is active, in which case remainder of SRAM is used). The DSP-C40 memory can be expanded by changing memory chips to as high as 6M x 32 of total SRAM. If you decide to modify the TMS320C4x source code to use SRAM for data processing or analog I/O buffers (see section 4.10, Texas Instruments TMS320C4x DSP Source Code Interface) instead of the default DRAM, we strongly suggest a 512k x 32 SRAM amount in order to fully enable all Hypersignal acceleration functions, allow continuous real-time functions to operate at full speed and capability, and allow long stimulus & response operations.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the DSP-C40 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

3.3 Atlanta Signal Processors, Inc. (ASPI)

3.3.1 ASPI Banshee (TMS320C30)

Basic Descriptive and Installation Details

Use a hardware designator of 'ASPIC30' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The Banshee contains the TMS320C30 32-bit floating-point DSP. When used with an AD16 analog I/O daughter board, is a two-channel system capable of 16-bit A/D and D/A. It is a full-length board with 16-bit ISA bus AT-style connector.

NOTE: **The Banshee** *must have an AD16 analog I/O daughter board installed for use with Hypersignal analog I/O functions.*

When the AD16 daughterboard is used, Intel 8254 timer-controlled sample rates are software selectable and range from 200 Hz to 200 kHz. However, the speed of current Hypersignal interrupt service routines running on the TMS320C30 limit maximum instrumentation rates to about 120 kHz. The use of the TMS320C30 DMA controller combined with the Banshee dual-port memory area allows Hypersignal to achieve 100 kHz single-channel hard disk transfer rates (25 MHz system with 25 msec avg. access-time hard drive).

Hypersignal uses a default I/O base address of 220H and a default memory base address of DC00H for the Banshee board. Note that you *can change* these addresses from the System Configuration menu, as is the case with most other boards. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS and MEM BASE ADDRESS fields, for details on setting these addresses if values other than 220H and DC00H are used. See below for more discussion on the choice of DC00H as the default memory base address.

NOTE: We highly suggest a memory base address of DC00H for the Banshee, instead of the D000H default used by ASPI. This will allow the board to work with network cards and/or "SuperVGA" video cards requiring extra "high memory". This will also preserve E000H-EFFFH as a possible EMS memory segment. Hypersignal software (and boards obtained through Signalogic) default to the DC00H value, requiring address jumpers to be changed on the board.

Also, be sure that you "block out" the region from DC00-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the Banshee.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C3x code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C30, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	100 kHz
1 channel Hard Disk Playback:	100 kHz
Max. 2-chn. Spectrum Analyzer Rate:	120 kHz
1024 pt FFT Accel. Disk File Rate:	20.3 frames/sec
Spectrum Analyzer Noise Floor:	87 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	34k x 32
Maximum Ext. Mem. Config:	130k x 32 (w/o daughterboard)
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes
Simultaneous S/H	yes
Typical Dynamic Range:	U/A
THD:	U/A
Voltage Range:	±3V
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	200 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes
Simultaneous S/H:	yes
Typical Dynamic Range:	U/A
THD:	U/A
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	200 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The Banshee is shipped standard with 32k x 32 of main memory (either program or data) and 2k x 32 of dual-port memory (used as data memory). Hypersignal uses less than 8k x 32 of the main memory for program, the remainder for analog buffers and processing areas, the first 1k x 32 of dual-port memory for intermediate DMA-driven transfers between host memory, and the last 1k x 32 of the dual-port area for shared variables between the host and the TMS320C30. A memory upgrade to 128k x 32 of main memory greatly improves stimulus & response operation by providing for larger stimuli and enables convolution/correlation acceleration. All other functions should perform equally well in either case.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the Banshee board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.4 AT&T

3.4.1 AT&T DSP32C Development Board

Basic Descriptive and Installation Details

Use a hardware designator of 'ATT32C' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The AT&T "DEV-xx" board is the AT&T manufactured development board containing the AT&T WE DSP32C 32-bit floating-point chip. "xx" refers to the two basic memory configurations in which the board is available: xx=16 for 16k x 32 of total memory, and xx=64 for 64k x 32 of total memory. Hypersignal will operate with either configuration (see Memory Usage discussion, below). It is a full-length board with 16-bit ISA bus AT-style connector.

One-channel analog I/O is provided by a 15-bit PCM codec with 10-bit linearity (AT&T part no. T7520, or other T752x family-member) to

facilitate development of algorithms that require analog signals. The codec provides built-in anti-aliasing and reconstruction filters for an expected sample rate of 8 kHz.

NOTE: The codec sampling rate is not programmable, but fixed proportionately to the processor oscillator rate—resulting in an 8 kHz sampling rate for a 50 MHz processor clock. *Thus, even though Hypersignal functions allow entry of sampling rates other than the fixed rate, actual programming of the sampling rate is not occurring.*

The initial default Hypersignal I/O base address for the AT&T DEV-xx is 300H, which is also the AT&T factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address. The value is marked in the I/O BASE ADDRESS field by "ATT".

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32C code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	19.2 Khz
1 channel Hard Disk Playback:	19.2 kHz
Max. 2-chn. Spectrum Analyzer Rate:	19.2 kHz
1024 pt FFT Accel. Disk File Rate:	UA
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	64k x 32
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes, also memory interface
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	none
Codec channels:	1, 15-bit T7520 μ-Law or A-Law
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±2.3V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, codec
Programmable Sampling Rate:	yes, codec
External Sample Clock:	no
Max. Sampling Rate:	8 kHz
Min. Sampling Rate:	8 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	none
Codec channels:	1, 15-bit T7520 μ-Law or A-Law
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±2.3V
Output Impedance:	UA
Reconstruction Filters:	yes, codec
Sin(x)/x Filters:	yes, codec
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The AT&T "DEV-xx" board standard memory configuration (xx=16) is 16k x 32 of zero wait-state external A area program memory and no external B area memory (total of 16 kword). The expanded configuration (xx=64) consists of installment of 48k x 32 of zero wait-state external B area memory (total of 64 kword).

Hypersignal uses less than 4k x 32 of the external A area for program storage, and either the rest of external A *or* the external B area (if installed) for analog buffers and file-based acceleration processing. Hypersignal will operate with either the -16 or -64 configuration, but the -64 configuration is strongly suggested to allow activation of all functions and maximum speed/size settings in other functions. Without the expanded memory, convolution/correlation acceleration will not be active, Spectrum Analyzer maximum 2-channel FFT size will be limited, maximum time data buffer sizes will not be as large as possible, and stimulus sizes in the stimulus and response function will be severely limited.

A daughterboard is also available for the AT&T DEV-xx board that can increase total memory size to 1M x 32, but the off-the-shelf configuration of Hypersignal DSP32C code makes no use of this memory at any time (see section 4.4, AT&T WE DSP32C DSP Source Code Interface, for details on customizing DSP32C code).

NOTE: Although the external B memory area is rated for 0 wait-state operation, Hypersignal DSP32C code sets the external B memory for 1 wait-state in order to allow accesses to the T7520 codec to function correctly.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the AT&T DEV-xx board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

Sampling rates for the AT&T development board can be entered in the high-level software but have no effect, since the board is fixed to an 8 kHz rate. If you provide an external clock input to the T7520, or change the DSP32C processor rate (on which the T7520 rate is proportionately based), you should then enter a matching number in the software.

3.5 Communications, Automation, and Control (CAC)

CAC hardware and associated software support utilities provide an excellent development environment for the AT&T DSP32 and DSP32C processors. Software development utilities such as the DSP3EMU emulator were developed in conjunction with AT&T engineers and provide reliable tools in addition to the basic AT&T assembler, linker, and software simulator. The DSP32-PC is a small, inexpensive board intended originally as an accelerator for PCs, although it does contain telecomm-quality analog I/O capability (8 or 12 kHz 13-bit codec). Hypersignal uses this board as an accelerator, a "DSP Coprocessor" for other acquisition boards with no DSP chip, and also supports the codec for real-time signal acquisition and generation functions. The DSP32C-AT, CAC's second-generation "big-brother" to the DSP32-PC, provides the much faster, improved DSP32C device, more memory, and 2 high-quality 16-bit analog I/O channels with 4th-order Butterworth input anti-aliasing and output reconstruction filters. The CAC-32C series of boards, CAC's third-generation hardware, also uses the DSP32C chip, but provides a wide variety of analog I/O options in the form of daughterboard modules.

3.5.1 CAC DSP32-PC (DSP32)

NOTE: This card can also serve as a "DSP Coprocessor" for supported acquisition boards that have no DSP chip, allowing Spectrum Analyzer and DSP Accelerator operation with these "analog-only" boards.

Basic Descriptive and Installation Details

Use a hardware designator of 'CAC32' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The CAC DSP32 board contains the AT&T WE DSP32 32-bit floating-point chip. In addition, it has a 1-channel 8-bit codec (coder-decoder) which log-expands (μ-Law or A-Law) A/D input to 13 bits and log-compands D/A output back to 8 bit D/A values, with its sampling frequency fixed at a rate proportional to the processor clock (usually 12 kHz, but can be slowed to 8 kHz). It is a 1/2 length board with 8-bit ISA bus XT-style connector.

The codec provides fully adequate anti-aliasing and reconstruction filters. Although a single-channel 8 kHz sampling rate capability may be inadequate for many systems, this board may still be desirable *in addition* to other hardware with greater analog capability, if only because of the inexpensive, accelerated floating-point complex FFT processing it can provide. In fact, this card can serve as a "DSP Coprocessor" for supported acquisition boards that have no DSP chip, allowing Spectrum Analyzer and DSP Accelerator operation.

Initial default Hypersignal I/O base address for the CAC board is 300H, which is also the CAC initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

NOTE: If you have a REV 1.1 or higher CAC board with 32k x 8 SRAM devices (static RAM, part no. xx256) then you must do the following because of a design change on the REV 1.1 CAC board: bend out pin 1 on all four (4) SRAM ICs. This still allows 16k x 32 of DSP32 memory to be accessed, but in addition, retains the board's ability to use the PIR register to communicate with the PC, which Hypersignal software and other AT&T application example software depends upon. Bending out pin 1s may not be enough; you may have to tie these pins high (+5V) in order to assure consistent system operation. Hypersignal software expects to use the DSP32 in memory mode 2, with all on-chip RAM located at F000H. On the DSP32-PC board, this means dip switch 6=ON, and switch 7=OFF.

NOTE: The codec sampling rate is not programmable, but fixed proportionately to the processor oscillator rate—resulting in either 8 kHz or 12 kHz sampling rates. Thus, even though Hypersignal functions allow entry of sampling rates other than the fixed rate, actual programming of the sampling rate is not occurring.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger entries in the Analog Conversion function are active (enter in range -4095 to 4095 for 13-bit A/D) and applied digitally, via DSP32 chip real-time software.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32, 160 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	12 kHz
1 channel Hard Disk Playback:	12 kHz
Max. 2-chn. Spectrum Analyzer Rate:	12 kHz
1024 pt FFT Accel. Disk File Rate:	9.0 frames/sec
Spectrum Analyzer Noise Floor:	62 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	32k x 32
Maximum Ext. Mem. Config:	32k x 32
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	none
Codec channels:	1, 13-bit T7500 μ-Law or A-Law
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±2V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, codec
Programmable Sampling Rate:	no
External Sample Clock:	no
Max. Sampling Rate:	12 kHz
Min. Sampling Rate:	8 kHz (use 240 nS DSP32 osc.)
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	none
Codec channels:	1, 13-bit T7500 μ-Law or A-Law
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±2V
Output Impedance:	UA
Reconstruction Filters:	yes, codec
Sin(x)/x Filters:	no
Programmable Sampling Rate:	no
External Sample Clock:	no
Max. Sampling Rate:	12 kHz
Min. Sampling Rate:	8 kHz (use 240 nS DSP32 osc.)
External TTL-Level Trigger:	no

Onboard Memory Usage

Some initial CAC DSP32 boards had only 8k x 32 of memory (either program or data); most have 16k x 32 of usable memory. Hypersignal uses 1k x 32 of the memory for program storage. Hypersignal will detect and operate with either amount of memory, but the 8k x 32 configuration will result in degraded operation (lower maximum FFT size and multichannel capability in Spectrum Analyzer operation); we highly recommend the 16k x 32 configuration.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the CAC DSP32 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

The stimulus & response function and snap-in real-time filtering function are not active for any DSP32 boards. Due to the difficulty in installing these functions on the DSP32 chip, which lacks a usable real-time interrupt structure, we have decided to not support these functions on DSP32 boards. Convolution acceleration is also not implemented on DSP32 boards. The DSP32C chip, on the other hand, is the "second-generation" DSP32, with many improvements; DSP32C-based boards have full functionality under Hypersignal and in fact boast some of the fastest performance numbers of all DSP chips currently supported.

3.5.2 CAC DSP32C-AT (DSP32C)

Basic Descriptive and Installation Details

Use a hardware designator of 'CAC32C' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

3.5.2 CAC DSP32C-AT (DSP32C)

The CAC DSP32C-AT board contains dual channel 16-bit analog I/O, including 4th-order Butterworth input anti-aliasing and output reconstruction filters with cutoff frequencies fixed at 25 kHz. It is a full-length board with 16-bit ISA bus AT-style connector.

The sampling rate of the analog system is fixed at 50 kHz, although an external signal may be applied to lower the sample rate to any frequency between DC and 50 kHz. It is important to keep this in mind when using Hypersignal; sampling rates entered by the user will be downloaded to DSP32C memory, but will *not affect actual sampling rates*. For accurate displays of the signal, the user must make sure that the entered rate matches the actual rate of the board (whether that be an externally supplied rate or the default 50 kHz).

Initial default Hypersignal I/O base address for the DSP32C-AT board is 300H, which is also the CAC initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

For proper operation of the DSP32C-AT board, please be aware that:

1) The DSP32C-AT must be installed in an AT-style (16-bit connector) bus to operate correctly.

2) On some machines, problems with the DSP32C-AT and Hypersignal have occurred when running with a high bus speed (above 8 MHz). The symptoms of this are machine lock-up while running a Hypersignal real-time function, and/or grossly incorrect data displays. The first correction to try is disabling the 80286 INSTRUCTION SET field in the System Configuration menu. The second correction is to slow the bus speed of your machine; some machines have a switch on the motherboard which will allow you to choose bus speed settings. Consult the documentation for your machine to see if this is the case.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	50 kHz
1 channel Hard Disk Playback:	50 kHz
Max. 2-chn. Spectrum Analyzer Rate:	50 kHz
1024 pt FFT Accel. Disk File Rate:	19 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	64k x 32
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	87 dB
THD:	UA
Voltage Range:	±3mV to ±3V; J4
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Butter4, FC25, RP
Programmable Sampling Rate:	no (fixed 50 kHz)
External Sample Clock:	yes
Max. Sampling Rate:	50 kHz
Min. Sampling Rate:	0
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	87 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, Butter4, FC25, RP
Sin(x)/x Filters:	no
Programmable Sampling Rate:	no (fixed 50 kHz)
External Sample Clock:	yes
Max. Sampling Rate:	50 kHz
Min. Sampling Rate:	0
External TTL-Level Trigger:	no

Onboard Memory Usage

The basic DSP32C-AT board contains 16k x 32 of zero wait-state external A area memory. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog data buffers and file-based acceleration processing. The DSP32C-AT memory can be expanded by changing memory chips to 64k x 32 of total external A memory. We strongly suggest the 64k x 32 amount in order to enable all Hypersignal acceleration functions and allow continuous real-time functions to operate at full speed and capability. For example, 12k x 32 of data memory is not enough to support 2-channel 4096 pt. FFT Spectrum Analyzer operation, but will allow smaller FFT sizes and/or 1-channel operation. Also, 64k x 32 of total memory will enable convolution/correlation acceleration, and greatly increase output stimulus sizes for stimulus & response operation.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the CAC DSP32C-AT board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

Sampling rates on the board are not programmable; see default and external sampling rate clock discussion above.

3.5.3 CAC CAC-32C / AC5 (DSP32C)

Basic Descriptive and Installation Details

Use a hardware designator of 'CAC32C' in the Hypersignal software System Configuration menu for the CAC-32C (also known as the "AC5-series") board with no analog I/O interface boards connected. Enter 'CAC32C-DBC' for the 2-channel (A/D-only) 16-bit sigma-delta "DBCS5339" daughterboard connected, 'CAC32C-DBD' for the 2-channel (A/D and D/A) 16-bit sigma-delta daughterboard connected, and 'CAC32C-COD' for the 15-bit linear T7525 codec daughterboard connected. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The CAC AC5 board contains the AT&T DSP32C 32-bit floating-point DSP, and allows connection to various small, low-profile analog I/O daughterboards. It is a 2/3 length board with 16-bit ISA bus AT-style connector.

The '-DBC' and '-DBD' analog I/O daughterboard options both contain dual channel 16-bit analog input, including full sigma-delta input anti-aliasing filters and linear input phase. Only the '-DBD' contains delta-sigma output operation, which includes interpolation filters and linear phase output. The sampling frequency for both of these options is fixed, in the sense that it is based on the crystal oscillator located on the daughterboard, which is usually a default value of 12.8 Mhz (50 kHz sampling frequency). The '-COD' analog I/O daughterboard option refers to a T7525 codec daughterboard, which provides 1 channel of 15-bit input and output, at a fixed sampling rate of 8 kHz (again based on the crystal oscillator located on the daughterboard).

It is important to keep the fixed sampling rate in mind when using Hypersignal; sampling rates entered by the user will be downloaded to DSP32C memory, but will *not affect actual sampling rates*. For accurate displays of the signal, the user must make sure that the entered rate matches the actual rate of the board (whether that be an externally supplied rate or the default 50 kHz).

Initial default Hypersignal I/O base address for the CAC-32C board is 300H, which is also the CAC initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

For proper operation of the CAC-32C board, please be aware that:

1) The CAC-32C can be installed in either an AT-style (16-bit connector) or PC-style (8-bit connector) bus to operate correctly.

2) On some machines, problems with the CAC-32C and Hypersignal have occurred when running with a high bus speed (above 8 MHz). The symptoms of this are machine lock-up while running a Hypersignal real-time function, and/or grossly incorrect data displays. The first correction to try is disabling the 80286 INSTRUCTION SET field in the System Configuration menu (a 'N' entry). The second correction is to slow the bus speed of your machine; some machines have a switch on the motherboard which will allow you to choose bus speed settings. Consult the documentation for your machine to see if this is the case.

Hardware Ratings and Software Throughput Specifications

NOTE: In the specifications below, values for the DBD, DBC, and COD daughterboards, respectively, are separated (when different) by a "/".

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	50 kHz
1 channel Hard Disk Playback:	50 kHz
Max. 2-chn. Spectrum Analyzer Rate:	50 kHz
1024 pt FFT Accel. Disk File Rate:	19 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	64k x 32
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits / 2, 16-bits / 1, 15-bits
Codec channels: 0 / 0 / 1	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	87 dB
THD:	UA
Voltage Range:	±0.36V to ±36V (±3.6V nom); TA
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD / SD / codec
Programmable Sampling Rate:	no (fixed 50 / 50 / 8 kHz)
External Sample Clock:	no (can change oscillator)
Max. Sampling Rate:	50 kHz
Min. Sampling Rate:	0
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits / 0 / 1, 15-bits
Codec channels:	0 / 0 / 1
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	87 dB
THD:	UA
Voltage Range:	±3.6V
Output Impedance:	UA
Reconstruction Filters:	yes, DS / no / codec
Sin(x)/x Filters:	no
Programmable Sampling Rate:	no (can change oscillator)
External Sample Clock:	no
Max. Sampling Rate:	50 kHz
Min. Sampling Rate:	0
External TTL-Level Trigger:	no

Onboard Memory Usage

The basic CAC-32C board contains 16k x 32 of zero wait-state external A area memory. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog data buffers and file-based acceleration processing. The CAC-32C memory can be expanded by changing memory chips to 64k x 32 of total external A memory. We strongly recom-

mend the 64k x 32 amount in order to enable all Hypersignal acceleration functions and allow continuous real-time functions to operate at full speed and capability. For example, 12k x 32 of data memory is not enough to support 2-channel 4096 pt. FFT Spectrum Analyzer operation, but will allow smaller FFT sizes and/or 1-channel operation. Also, 64k x 32 of total memory will enable convolution/correlation acceleration, and greatly increase output stimulus sizes for stimulus & response operation.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the CAC board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

Sampling rates on the board are not programmable; see default and external sampling rate clock discussion above.

3.5.4 CAC QUANTUMdsp (DSP3210)

Basic Descriptive and Installation Details

Use a hardware designator of 'CAC3210' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The QUANTUMdsp contains the 32-bit floating-point AT&T DSP3210 DSP chip, along with two (2) 16-bit analog I/O channels and a DAA telephone line interface. It is a 2/3 length AT-style board with 16-bit ISA bus interface.

Analog input and output uses sigma-delta A/D and delta-sigma D/A converters, which provide programmable sampling rates, built-in anti-aliasing and reconstruction filters that track exactly with the programmed sampling rates, and linear phase. One Crystal Semiconductor CS4216 stereo audio codec device is used for A/D and D/A conversion. This device is rated at about 80 dB of dynamic range. The software-selectable sampling rates on this board must be one of 8 discrete values in a range from 8 to 48 kHz allowed by the 24.576 MHz onboard crystal oscillator. Possible values are 8, 16, 24, 32, and 48 kHz, as well as audio rates 11.025, 22.05 and 44.1 kHz. Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP3210 Source Code Interface, it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion. Also, the sampling rate oscillator on the QUANTUMdsp board can be replaced with a different value to achieve desired real-time sampling rates with optimum performance (no extra signal processing overhead).

The telephone line interface is capable of being used for V32bis MODEM, Class I, Group III fax (V.29 - 9600 baud), voice mail, and answering-machine applications. The interface also supports the addition of an 8 ohm in-progress monitor speaker and ANI CALLER ID.

Initial default Hypersignal I/O base address for the DSP32C-AT board is 360H, which is also the CAC initial factory default. You must change the System Configuration menu I/O BASE ADDRESS field for the CAC subfield before executing QUANTUMdsp board functions, because the standard Hypersignal default value for this subfield is set to 300H for other CAC boards. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

An unusual feature of the QUANTUMdsp board is that it contains an EEPROM that allows space for application specific data. Default data in the EEPROM contains serial number and date of manufacture.

Another unusual feature is that all onboard external memory is DRAM, as opposed to the traditional SRAM used on DSP boards. There are two major ramifications to the use of DRAM:

1) DSP execution is significantly slower. DSP3210 access to DRAM require from 4-6 wait-states. Since the maximum sampling rate is only 48 kHz, most Hypersignal functions are unaffected, although 2-channel stimulus & response at rates above 24 kHz may not be possible, depending on the speed of the host machine. However, processing-intensive functions such as real-time convolution will not run at sampling rates above 12-16 kHz, even though such a large amount of memory would initially seem to make the board suitable for these type of operations.

2) Cost is reduced. The price of the QUANTUMdsp is quite low for a DSP board containing a 32-bit floating-point processor, 512k x 32 of memory, analog I/O, and a telephone interface.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock or non-standard crystal oscillators and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order

to maintain correct time units display and usage. If you change oscillator values, then you will need to make minor modifications to the DSP3210 source code in order to program the sampling rate latch correctly (because the built-in Hypersignal programming is disabled by the '-EXT' selection). See section 4.9, AT&T DSP3210 DSP Source Code Interface.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP3210 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP3210, 72.7 nsec
Data Format, Wordlength:	floating-point, 32-bit
2 channel Hard Disk Record:	48 kHz
2 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	48 kHz
1024 pt FFT Accel. Disk File Rate:	15 frames/sec
Spectrum Analyzer Noise Floor:	80 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	512k x 32 (DRAM)
Maximum Ext. Mem. Config:	512k x 32 (DRAM)
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes (proc. expansion connector)
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	80 dB
THD:	UA
Voltage Range:	±1.4V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P8
External Sample Clock:	no
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	8 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	80 dB
THD:	UA
Voltage Range:	±2.98V
Output Impedance:	UA
Reconstruction Filters:	yes, DS
Sin(x)/x Filters:	yes, DS
Programmable Sampling Rate:	yes, P8
External Sample Clock:	no
Max. Sampling Rate:	48 kHz 16-bit
Min. Sampling Rate:	7.5 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The QUANTUMdsp board contains 512k x 32 of external DRAM. This memory runs at between 4 and 6 wait-states, depending on the processor context when a memory access is made. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog I/O data buffers, file-based math/DSP function processing (acceleration), instrumentation and measurement calculations, and stimulus storage. The large amount of memory fully enables all accelerated math/DSP functions and spectrum analyzer measurements, and allows stimlus & response operations with very long stimuli. The relatively slow

speed of the DRAM limits acceleration performance, and reduces maximum stimulus & response sampling rates to the 20 - 24 kHz range.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the QUANTUMdsp board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

3.6 Data Translation

Hypersignal supports all Data Translation 282x-series hardware. Generally, this means 2821 and 2823 boards, and related boards with the same host interface, but different analog features and capabilities. Significant features of Data Translation hardware include fast continuous transfer rates due to the dual-channel host DMA interface, programmable hardware gain registers, external trigger option, external clock option, and flexible, random-access channel scanning.

3.6.1 Data Translation DT2821 (No DSP Chip)

NOTE: Even though this board has no DSP chip, Spectrum Analyzer and DSP Accelerator operation can still be achieved via a separate, inexpensive "DSP Coprocessor" board. See section 3.5.1, above, for a description of one such board. See the product catalog for pricing information and further details.

Basic Descriptive and Installation Details

Use a hardware designator of 'DT2821' in the Hypersignal software System Configuration menu (except for DSP ACCELERATOR field). It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The DT2821, DT2821-F, and DT2821-G all have 8 differential or 16 single-ended 12-bit input channels and 2 12-bit output channels and can sample at software-programmable rates from 0 up to 50 kHz, 150 kHz and 250 kHz, respectively. None of the DT2821 series boards contain anti-aliasing or reconstruction filters—see section 2.0, HARDWARE CONSIDERATIONS, above, for possible remedies to this problem. These boards are full-length board with 16-bit ISA bus AT-style connectors.

Initial default Hypersignal I/O base address for DT2821 hardware is 240H, which is also the Data Translation initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

NOTES: Hypersignal software expects all DT2821 hardware to have input A/D coding format jumper options *set to 2's complement, not offset binary.*

You may note from the specifications listed below that continuous disk output on the DT2821 is slower than input. This results from the fact that the DT2821 series output *is hardwired to offset binary* (all values positive, as opposed to 2's complement). Since DSP analysis is generally expected to be performed and displayed in 2's complement representation, each output point must be handled individually by the host 80x86 processor in order to add an offset. This precludes a pure DMA transfer and slows the overall D/A output rate. You can effectively cancel the automatic output-offset addition by entering -2048 as the OUTPUT OFFSET in the Analog Conversion function, regaining the maximum transfer rate.

The actual, overall sampling rate being used by the DT2821 board for multichannel acquisition is *the entered rate multiplied by the number of channels in operation.* If this actual rate exceeds the maximum rate specified for the board, then results will be unpredictable.

Output is available on either channels 0 or 1, or alternating output between the two channels (multichannel output). Alternating output on two channels amounts to dual, or "stereo", output if a file with interleaved channel data (see multichannel acquisition discussion in Users Manual section 6.2, Analog Conversion, under CHANNEL LIST field) is used as the input waveform file. Thus, either a previously acquired stereo waveform data file or a user-prepared file must be available.

3.6.1 Data Translation DT2821 (No DSP Chip)

For DT2821 hardware, only limited input signal gain is allowed, but this gain is implemented in analog hardware and is thus a "true" gain as opposed to DSP chip software scaling, which is used by most boards with an onboard DSP chip. DT2821 hardware input gain is programmable to discrete levels of 1, 2, 4 or 8. Entry of output gain values is ignored. The current gain value applies to all input channels in operation. Gain values are set directly in the Analog Conversion function and in Volts in the INPUT RANGE field in the Digital Oscilloscope and Spectrum Analyzer functions. In these latter two functions, you should set the INPUT RANGE field to 10 (±10 Volts full-scale), 5, 2.5, or 1.25.

The TRIGGER MODE, LEVEL field in the Analog Conversion function can be set for either host software-initiated trigger ('INT') or external TTL-level initiated trigger ('EXT'), but no data threshold LEVEL can be specified. The external trigger applies to both input and output operation. In the Digital Oscilloscope and Spectrum Analyzer functions you can use the standard trigger controls ('+' and '-' keys) to enable/disable the external trigger option.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking and enable the external clock input on the DT2821 backplane. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset entries in the Analog Conversion function are active (enter in range -2047 to 2047 for 12-bit A/D) and applied digitally, via 80x86 assembly language. Use of offsets can greatly affect speed of transfer—see the output offset discussion above.

Hardware Ratings and Software Throughput Specifications

NOTE: In the hardware specifications below, values are given for the DT2821, DT2821-F, and DT2821-G. These values are separated (when different) by a "/".

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	None
Data Format, Wordlength:	None
1 channel Hard Disk Record:	50/150/150 kHz
1 channel Hard Disk Playback:	50/120/120 kHz
Max. 2-chn. Spectrum Analyzer Rate:	50/75/125 kHz
1024 pt FFT Accel. Disk File Rate:	N/A
Spectrum Analyzer Noise Floor:	79/79/79 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	N/A
Maximum Ext. Mem. Config:	N/A
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes
Host Data Transfer Method:	DMA

A/D SPECIFICATIONS

Number of linear PCM channels:	8DI,12-bits / 8DI,12 / 8DI,12
Codec channels:	none
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	75/75/75 dB
THD:	.05%
Voltage Range:	±1.25V to ±10V; P4
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	50/150/250 kHz - 1CH
Min. Sampling Rate:	0.467 Hz
External TTL-Level Trigger:	yes
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 12-bits OB / 2, 12 OB / 2, 12 OB
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	75/75/75 dB
THD:	.05%
Voltage Range:	±1.25V to ±10V; J4
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	50/150/250 kHz
Min. Sampling Rate:	0.467 Hz
External TTL-Level Trigger:	yes

Onboard Memory Usage

No onboard memory.

Additional Notes About Hypersignal Software Operation

1) Digital Oscilloscope and Spectrum Analyzer triggering operation (see discussion above).

2) The "snap-in" real-time digital filtering, stimulus & response, and "monitor mode" (input loopbacked to output) functions are disabled—these functions are only enabled for hardware with an onboard DSP chip.

3) Stereo output for a CHANNEL LIST entry of '1,0' does not switch the physical channel output from a CHANNEL LIST entry of '0,1'. This is due to a limitation on the DT2821 board.

3.6.2 Data Translation DT2823 (No DSP Chip)

NOTE: Even though this board has no DSP chip, Spectrum Analyzer and DSP Accelerator operation can still be achieved via a separate, inexpensive "DSP Coprocessor" board. See section 3.5.1, above, for a description of one such board. See the product catalog for pricing information and further details.

Basic Descriptive and Installation Details

Use a hardware designator of 'DT2823' in the Hypersignal software System Configuration menu (except for DSP ACCELERATOR field). It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The DT2823 has 4 differential or 8 single-ended 16-bit input channels and 2 16-bit output channels and can sample at software-programmable rates from 0 up to 100 kHz. None of the DT2823 series boards contain anti-aliasing or reconstruction filters—see section 2.0, HARDWARE CONSIDERATIONS, above, for possible remedies to this problem. The DT2823 is a 2/3 length board with 16-bit ISA bus AT-style connector.

Initial default Hypersignal I/O base address for DT2823 hardware is 240H, which is also the Data Translation initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

NOTES: Hypersignal software expects all DT2823 hardware to have both input A/D and output D/A coding format jumper options *set to 2's complement, not offset binary.*

The actual, overall sampling rate being used by the DT2823 board for multichannel acquisition is *the entered rate multiplied by the number of channels in operation.* If this actual rate exceeds the maximum rate specified for the board, then results will be unpredictable.

Output is available on either channels 0 or 1, or alternating output between the two channels (multichannel output). Alternating output on two channels amounts to dual, or "stereo", output if a file with interleaved channel data (see multichannel acquisition

discussion in Users Manual section 6.2, Analog Conversion, under CHANNEL LIST field) is used as the input waveform file. Thus, either a previously acquired stereo waveform data file or a user-prepared file must be available.

For DT2823 hardware, only limited input signal gain is allowed, but this gain is implemented in analog hardware and is thus a "true" gain as opposed to DSP chip software scaling, which is used by most boards with an onboard DSP chip. DT2823 hardware input gain is programmable to discrete levels of 1, 2, 4 or 8. Entry of output gain values is ignored. The current gain value applies to all input channels in operation. Gain values are set directly in the Analog Conversion function and in Volts in the INPUT RANGE field in the Digital Oscilloscope and Spectrum Analyzer functions. In these latter two functions, you should set the INPUT RANGE field to 10 (±10 Volts full-scale), 5, 2.5, or 1.25.

The TRIGGER MODE, LEVEL field in the Analog Conversion function can be set for either host software-initiated trigger ('INT') or external TTL-level initiated trigger ('EXT'), but no data threshold LEVEL can be specified. The external trigger applies to both input and output operation. In the Digital Oscilloscope and Spectrum Analyzer you can use the standard trigger controls to enable/disable the external trigger option.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking and enable the external clock input on the DT2823 backplane. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via 80x86 assembly language. However, use of non-zero offsets can greatly reduce the maximum rate of continuous transfer to/from disk. This results from the host 80x86 processor handling each point individually in order to add an offset, precluding a pure DMA transfer and slowing the overall input/output rate.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	None
Data Format, Wordlength:	None
1 channel Hard Disk Record:	100 kHz
1 channel Hard Disk Playback:	100 kHz
Max. 2-chn. Spectrum Analyzer Rate:	50 kHz
1024 pt FFT Accel. Disk File Rate:	N/A
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	N/A
Maximum Ext. Mem. Config:	N/A
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes
Host Data Transfer Method:	DMA

A/D SPECIFICATIONS

Number of linear PCM channels:	4DI, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	87 dB
THD:	.05%
Voltage Range:	±1.25V to ±10V; P4
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	100 kHz - 1CH
Min. Sampling Rate:	0.467 Hz
External TTL-Level Trigger:	yes
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	87 dB
THD:	.05%
Voltage Range:	±1.25V to ±10V; J4
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	100 kHz
Min. Sampling Rate:	0.467 Hz
External TTL-Level Trigger:	yes

Onboard Memory Usage

No onboard memory.

Additional Notes About Hypersignal Software Operation

1) Digital Oscilloscope and Spectrum Analyzer triggering operation (see discussion above).

2) The "snap-in" real-time digital filtering, stimulus & response, and "monitor mode" (input loopbacked to output) functions are disabled—these functions are only enabled for hardware with an onboard DSP chip.

3) Stereo output for a CHANNEL LIST entry of '1,0' does not switch the physical channel output from a CHANNEL LIST entry of '0,1'. This is due to a limitation on the DT2823 board.

3.7 Digital Audio Labs

Digital Audio Labs hardware was designed specifically for audio and acoustic applications, with special attention paid to phase linearity and dynamic range. The DRB1 and DRB2 are essentially the same, except the DRB2 uses sigma-delta converters for analog input.

3.7.1 Digital Audio Labs DRB1 (No DSP Chip)

NOTE: Even though this board has no DSP chip, Spectrum Analyzer and DSP Accelerator operation can still be achieved via a separate, inexpensive "DSP Coprocessor" board. See section 3.5.1, above, for a description of one such board. See the product catalog for pricing information and further details. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

Basic Descriptive and Installation Details

Use a hardware designator of 'DRB1' in the Hypersignal software System Configuration menu (except for DSP ACCELERATOR field).

The DRB1 is a 16-bit stereo board with sampling rates programmable to 32, 44.1, and 48 kHz. Input is from two simultaneous sample-and-hold 16-bit successive-approximation A/D converters, with 6th-order Butterworth anti-aliasing filters. Output is from two simultaneous 18-bit D/A converters (which use an 8x over-sampling technique to achieve the extra 2-bit resolution), with 4th-order Butterworth reconstruction filters and post-filter phase-compensation to ensure output phase linearity. The DRB1 is a full-length board with 8-bit ISA bus XT-style connector.

The default I/O BASE ADDRESS for the DRB1 board is 1C8H. This value cannot be changed in the System Configuration menu, but can be changed by editing the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly (with an ASCII text-editor). See the "CONFIG.DOC" file for an example of how to access the "HSMACRO.CNF" (or "HSACOUS.CNF") file and the exact location of the DRB1 I/O base address.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a

matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via 80x86 assembly language. However, use of non-zero offsets can greatly reduce the maximum rate of continuous transfer to/from disk. This results from the host 80x86 processor handling each point individually in order to add an offset, precluding a pure DMA transfer and slowing the overall input/output rate.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	None
Data Format, Wordlength:	None
1 channel Hard Disk Record:	48 kHz
1 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	48 kHz
1024 pt FFT Accel. Disk File Rate:	N/A
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	N/A
Maximum Ext. Mem. Config:	N/A
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	DMA

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	0
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	.005%
Voltage Range:	±1.73V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Butter6
Programmable Sampling Rate:	yes, P3
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	32 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	95 dB
THD:	.003%
Voltage Range:	±1.73V
Output Impedance:	UA
Reconstruction Filters:	yes, Butter4
Sin(x)/x Filters:	yes
Programmable Sampling Rate:	yes, P3
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	32 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

There is no memory on the DRB1 board.

Additional Notes About Hypersignal Software Operation

1) Triggering in the Digital Oscilloscope and Spectrum Analyzer functions is disabled since this board does not have either programmable-level voltage comparators nor or an external TTL-level trigger option.

2) The "snap-in" real-time digital filtering and stimulus & response functions are disabled—these functions are only enabled for hardware with an onboard DSP chip. The "monitor mode" function (input loopbacked to output) is enabled.

3) Only stereo-channel operation is allowed in the Analog Conversion and D/A output from display functions. Mono (single-channel) capability is enabled in the instrument display functions only.

3.7.2 Digital Audio Labs DRB2 (No DSP Chip)

NOTE: Even though this board has no DSP chip, Spectrum Analyzer and DSP Accelerator operation can still be achieved via a separate, inexpensive "DSP Coprocessor" board. See section 3.5.1, above, for a description of one such board. See the product catalog for pricing information and further details.

Basic Descriptive and Installation Details

Use a hardware designator of 'DRB2' in the Hypersignal software System Configuration menu (except for DSP ACCELERATOR field). It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The DRB2 is a 16-bit stereo board with sampling rates programmable to 32, 44.1, and 48 kHz. Input is from two simultaneous sample-and-hold 16-bit sigma-delta A/D converters, which provide near-perfect anti-aliasing filters through onchip digital FIR filtering techniques, and exactly linear phase. Output is from two simultaneous 18-bit D/A converters (which use an 8x over-sampling technique to achieve the extra 2-bit resolution), with 4th-order Butterworth reconstruction filters and post-filter phase-compensation to ensure output phase linearity. The DRB2 is a 2/3 length board with 8-bit ISA bus XT-style connector.

The default I/O BASE ADDRESS for the DRB2 board is 1C8H. This value cannot be changed in the System Configuration menu, but can be changed by editing the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly (with an ASCII text-editor). See the "CONFIG.DOC" file for an example of how to access the "HSMACRO.CNF" (or "HSACOUS.CNF") file and the exact location of the DRB2 I/O base address.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via 80x86 assembly language. However, use of non-zero offsets can greatly reduce the maximum rate of continuous transfer to/from disk. This results from the host 80x86 processor handling each point individually in order to add an offset, precluding a pure DMA transfer and slowing the overall input/output rate.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	None
Data Format, Wordlength:	None
1 channel Hard Disk Record:	48 kHz
1 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	48 kHz
1024 pt FFT Accel. Disk File Rate:	N/A
Spectrum Analyzer Noise Floor:	95 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	N/A
Maximum Ext. Mem. Config:	N/A
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	DMA

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits (SD)
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	92 dB
THD:	.003%
Voltage Range:	±1.73V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P3
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	32 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	95 dB
THD:	.003%
Voltage Range:	±1.73V
Output Impedance:	UA
Reconstruction Filters:	yes, Butter4
Sin(x)/x Filters:	yes
Programmable Sampling Rate:	yes, P3
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	32 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

There is no memory on the DRB2 board.

Additional Notes About Hypersignal Software Operation

1) Triggering in the Digital Oscilloscope and Spectrum Analyzer functions is disabled since this board does not have either programmable-level voltage comparators nor or an external TTL-level trigger option.

2) The "snap-in" real-time digital filtering and stimulus & response functions are disabled—these functions are only enabled for hardware with an onboard DSP chip. The "monitor mode" function (input loopbacked to output) is enabled.

3.8 Domain Technologies

Domain Technologies produces an enhanced version of the Texas Instruments EVM TMS320C30 board, with the following major differences:

1) TMS320C31 DSP used instead

2) Standard external onboard SRAM is 32k x 32 (vs. 16k x 32 on EVM), and can be expanded to 128k x 32

3) Option for AIC analog I/O is clocked with an additional, independent 5.184 MHz oscillator to provided standard baud rates

4) A 16-bit ISA bus interface is used for higher performance

3.8.1 Domain Technologies DSPCard (TMS320C31)

Basic Descriptive and Installation Details

Use a hardware designator of 'DOMC31' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The Domain Technologies DSPCard contains the Texas Instruments TMS320C31 32-bit floating-point DSP, and one analog I/O channel. It is a 2/3 length board with 16-bit ISA bus AT-style connector.

The DSPCard contains a TLC32044 AIC connected to the TMS320C31 serial channel. The TLC32044 can be programmed from Hypersignal for different sampling rates and one of two different input gains. The TLC32044 AIC (analog interface chip) provides one channel of 14-bit A/D and D/A. The TLC32044 device allows the sampling rate to be programmed, 3 levels of input gain to be selected, and contains anti-aliasing and reconstruction bandpass filters that track with the programmed sampling rate. The sampling rate and gain are controllable from Hypersignal menus. One should be aware that programming of the sampling rate is not exact, because the TLC3204x device uses two 6-bit integer counters as "divisors" into a master clock frequency, which allows only a discrete number of possibilities. Normally, input levels can be selected to be ±1.5V, ±3V, and ±6V. Also, the highpass portion of the bandpass filters can be selectively disabled (although this must be done through the Hypersignal configuration file as opposed to data-entry menus). For TLC32044 devices, a sin(x)/x D/A output filter is also incorporated.

NOTE: Before running Hypersignal with the Domain DSPCard, you must download a small host-interface program using the Domain Technologies LDR_C30 loader. To do this, locate the LDR_C30.EXE file on the Domain "Development System for Texas Instruments 320C3x" disk, and enter the following DOS command:

LDR_C30 DOM_INIT.OUT <car ret>

The "DOM_INIT.OUT" file is contained on Hypersignal distribution diskette #2 and will be automatically copied in a standard installation to the Hypersignal subdirectory on the hard drive. You must enter the ".OUT" extension in the command.

If you plan to make any changes to the TMS320C3x source code running on the DSPCard, please consult section 4.5, Texas Instruments TMS320C3x DSP Source Code Interface, for notes on ensuring that the "DOM_INIT.OUT" file is kept up to date along with any other changes.

Hypersignal uses a default I/O base address of 240H for the DSPCard board. Note that you *cannot* change this address from the System Configuration menu, as is the case with other boards. To change the base address, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal. See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the DSPCard I/O base address in the "HSMACRO.CNF" (or "HSACOUS.CNF") file.

By editing the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly (outside of Hypersignal, with a text-editor), TLC32044 parameters such as anti-alias and reconstruction filter enable/disable, synchronous I/O, AUX input, etc. can be programmed differently from the Hypersignal default. See the "CONFIG.DOC" file (on distribution diskette #2) for details both on the default values and on the location and format of the AIC control word in the configuration file. The rated maximum sampling rate for the TLC32044 is about 20 kHz, although some devices will operate adequately at rates up to 30 kHz.

In the Digital Oscilloscope and Spectrum Analyzer functions, an INPUT RANGE field setting of 3 (±3 Volts) will allow accurate measurement of input voltages, but you should attempt to keep input below ±2.5V to avoid any non-linearities and distortion. If you change the INPUT RANGE field to 1.5, you will increase the gain of the AIC chip to 2 (input at ±1.5V full-scale). The TLC32044 device on the DSPCard has been hard-wired for single-ended analog input operation, and is not set for differential-ended operation, which eliminates the ±6V full-scale possibility.

NOTE: Default Hypersignal operations expects a 32 MHz TMS320C31 clock rate, which is used to derive default AIC sampling rates. Any other value should be entered in the System Configuraton menu (entry of '-NN' after the hardware designator; see Users Manual section 2.5.2, Initializing the System Configuration Menu, for details). An similar entry of '-EXT' will disable built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the various software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and readouts.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -8191 to 8191 for 14-bit A/D) and applied digitally, via TMS320C3x code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C30, 62.5 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	19.2 kHz
1 channel Hard Disk Playback:	19.2 kHz
Max. 2-chn. Spectrum Analyzer Rate:	19.2 kHz
1024 pt FFT Accel. Disk File Rate:	UA
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	32k x 32
Maximum Ext. Mem. Config:	128k x 32
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x AIC
Codec channels:	none
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±1.5V to ±3V; P2
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby7, JB (AIC)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x AIC
Codec channels:	none
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±6V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby7, JB (AIC)
Sin(x)/x Filters:	yes, AIC (TLC32044)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The standard external SRAM configuration of the DSPCard is 32k x 32 (either program or data); zero wait-state. SRAM can be expanded to 128k x 32. Hypersignal uses less than 8k x 32 of the memory for program storage. The remaining data memory is used for analog I/O buffers, spectral analysis, DSP/math function processing, stimulus storage, etc. The standard 32k x 32 memory amount limits the size of stimuli in stimulus & response operations, limits the maximum file-based FFT acceleration and Spectrum Analyzer FFT size to 1024, and disables convolution/correlation acceleration. The 128k x 32 amount avoids these limitations, and is strongly recommended.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the DSPCard in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.9 DSP Research

3.9.1 DSP Research TIGER 30 (TMS320C30)

Basic Descriptive and Installation Details

Use a hardware designator of 'DRIC30' in the Hypersignal software System Configuration menu for standard dual TLC32044 operation, and 'DRIC30-BB71' for operation with the BB71 external analog I/O unit. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The TIGER 30 board contains the Texas Instruments TMS320C30 32-bit floating-point DSP, and two analog I/O channels. It is a full-length board with 16-bit ISA bus AT-style connector.

The default TIGER 30 board contains dual TLC32044 AICs, which can be programmed from Hypersignal for different sampling rates and one of two different input gains. The TLC32044 AICs (analog interface chip) each provide one channel of 14-bit A/D and D/A, and are each connected to one of the two TMS320C30 serial port channels. The TLC32044 device allows the sampling rate to be programmed, 3 levels of input gain to be selected, and contains anti-aliasing and reconstruction bandpass filters that track with the programmed sampling rate. The sampling rate and gain are controllable from Hypersignal menus. One should be aware that programming of the sampling rate is not exact, because the TLC3204x device uses two 6-bit integer counters as "divisors" into a master clock frequency, which allows only a discrete number of possibilities. Normally, input levels can be selected to be ±1.5V, ±3V, and ±6V. Also, the highpass portion of the bandpass filters can be selectively disabled (although this must be done through the Hypersignal configuration file as opposed to data-entry menus). For TLC32044 devices, a sin(x)/x D/A output filter is also incorporated.

By editing the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly (outside of Hypersignal, with a text-editor), TLC32044 parameters such as anti-alias and reconstruction filter enable/disable, synchronous I/O, AUX input, etc. can be programmed differently from the Hypersignal default. See the "CONFIG.DOC" file (on distribution diskette #2) for details on the default values, location, and format of the AIC control word in the configuration file. The rated maximum sampling rate for the TLC32044 is about 20 kHz, although some devices will operate adequately at rates up to 30 kHz.

Optionally, Hypersignal also supports a TIGER 30 board with the TLC32044 AICs removed, where the serial ports are used instead to communicate to the external "BB71" 4-channel analog I/O unit. The BB71 contains 4 16-bit input channels and 4 16-bit output channels. The sampling rate is fixed, but can be set to one of 256 different sampling rates via an external switch (up to a maximum of 200 kHz for the input channels, and 500 kHz for the output channels). The BB71 also offers better than 90 dB of spurious-free dynamic range, and optional audio (20 kHz cutoff frequency) 6-pole Bessel filters on all channels (which can be jumper-enabled, and also resistor-programmed to be of type Butterworth, Chebyshev, or elliptical!). *Although the BB71 is capabable of sampling at rates beyond 120 kHz, it is unlikely that Hypersignal and/or the TMS320C30 will be able to stream data continuously to disk at these rates, due to the time required for simultaneous real-time interrupt processing and continuous data transfer between the host PC and TIGER 30 dual-port memory.*

Hypersignal uses a default I/O base address of 220H and a default memory base address of DC00H for the TIGER 30 board. Note that you *cannot* change these address from the System Configuration menu, as is the case with other boards. To change the base addresses, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal. See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the TIGER 30 board I/O and memory base addresses in the "HSMACRO.CNF" (or "HSACOUS.CNF") file. Note also that the memory base address is never actually set via jumpers or switches on the board (as is the I/O base address), but is instead written to registers on the board via high-level Hypersignal software.

NOTE: We highly suggest you not change the default value of DC00H for memory base address. This value will allow the board to work with network cards and/or "SuperVGA" video cards requiring extra "high memory". This value also preserves E000H-EFFFH as a possible EMS memory segment. Hypersignal software (and the board, if obtained through Signalogic) always defaults to the DC00H value.

Also, be sure that you "block out" the region from DC00-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the TIGER 30.

NOTE FOR TLC32044 USE: In the Digital Oscilloscope and Spectrum Analyzer functions, an INPUT RANGE field setting of 6 (±6 Volts) will allow accurate measurement of input voltages, but you should attempt to keep input below ±4V to avoid any non-linearities and distortion. If you change the INPUT RANGE field to 3, you will increase the gain of the AIC chip to 2 (input at ±3V full-scale). The default setting for the TLC32044 devices on the TIGER 30 board is differential input operation.

An entry of '-NN' after the entry for this board in the System Configuration menu will force Hypersignal sampling rate programming to be based on the value NN. NN should be specified in MHz; no entry sets the default value of 33.21 MHz (the processor clock rate used by a standard TIGER 30 board). For 40 MHz TIGER 30 boards, you can enter '-40' after the System Configuration designators to inform the software about the faster speed. For TLC32044 AIC operation, the processor clock is actually divided by four before input to the TLC32044s; this division by 4 is accounted for by the software. On the TIGER 30 you can also install an "external oscillator" and set a jumper to enable it—in which case you should enter a value in the System Configuration menu that is 4x the rate of the external osciallator. Note that for BB71 use, a clock rate value entered in the System Configuration menu will mean nothing, since sampling rates for the BB71 are set by external switches located on the back of the BB71 unit.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

NOTES FOR BB71 USE: The above advice for external clock use applies to the BB71 since the rate is basically fixed (switch-selectable). It is important to keep the fixed sampling rate in mind when using Hypersignal; sampling rates entered by the user are downloaded to the TMS320C30, but will *not affect actual sampling rates.* **For accurate waveform displays, the user must make sure that the entered rate matches the actual switch setting.**

Second, note that the switch settings, as explained in the BB71 manual, can be confusing. Basically, you should calculate the value for N as explained in the manual, and then set the switch using N as a binary number and the switch positions as bits of the binary number. Finally, note that the master oscillator inside the BB71 can be changed to allow different sampling rates. The default value for this clock is 11.2896 MHz, which must be increased to 14.4 MHz to allow the A/D converters to run at their maximum rate of 200 kHz (apparently the switch can never be set below N=17).

Third, note that the AIC chips are installed in standard shipments of the TIGER 30 board must be removed before you can connect the BB71. The AICs are square (about 1.27 cm each side) ICs that fit within two "flat-pack" IC holders located near the backplane of the board. Remove the AICs after power to the host computer has been turned off, and with the TIGER 30 removed from the computer. The BB71 is expected to be connected to the top connector on the TIGER 30 board (connector located away from the edge connector for the host I/O bus).

Finally, note that Hypersignal supports both revision "A" and revision "B" of the BB71. A System Configuration entry of 'DRIC30-BB71B' specifies the revision B units; an entry of 'DRIC30-BB71' also defaults to revision B. An entry of 'DRIC30-BB71A' specifies revision A. DSP Research created the revision B unit to reverse the polarity of one of the clock signals carried in the interface cable, in order to improve noise resistance and length of the cable.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C30 code.

Hardware Ratings and Software Throughput Specifications

NOTE: In the specifications below, values for the TLC32044 and BB71 analog I/O options are separated (when different) by a "/".

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C30, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	19.2 kHz / 100 kHz
1 channel Hard Disk Playback:	19.2 kHz / 100 kHz
Max. 2-chn. Spectrum Analyzer Rate:	19.2 kHz / 120 kHz
1024 pt FFT Accel. Disk File Rate:	20.3 frames/sec
Spectrum Analyzer Noise Floor:	UA / 90 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	32k x 32
Maximum Ext. Mem. Config:	128k x 32
Ext. Sync Serial Interface:	yes (two ports available for
Ext Async. Serial Interface:	sync/async if analog I/O not used)
External Parallel Interface:	no
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 14-bit TLC3204x AIC / 2, 16-bit
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	70 dB / 90 dB
THD:	UA
Voltage Range:	±1.5V to ±6V; P3 / ±2.75V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby7, JB (AIC) / yes, BESSEL6, JB
Programmable Sampling Rate:	yes / no (DIP switch-selectable)
External Sample Clock:	no / yes
Max. Sampling Rate:	19.2 kHz / 200 kHz
Min. Sampling Rate:	1 kHz / clk/1024
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 14-bit TLC3204x AIC / 2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	70 dB / UA
THD:	UA
Voltage Range:	±6V / ±3V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby7, JB (AIC) / yes, BESSEL6, JB
Sin(x)/x Filters:	yes, AIC (TLC32044) / no
Programmable Sampling Rate:	yes / no (DIP switch-selectable)
External Sample Clock:	no / yes
Max. Sampling Rate:	19.2 kHz / 500 kHz
Min. Sampling Rate:	1 kHz / clk/1024
External TTL-Level Trigger:	no

Onboard Memory Usage

The TIGER 30 is shipped standard with 32k x 32 of main, dual-port memory (either program or data). Hypersignal uses less than 8k x 32 of the dual-port memory for program, and the remainder for analog data buffers and processing areas. A memory upgrade to 128k x 32 of dual-port memory greatly improves stimulus & response operation by providing for larger stimuli and enables convolution/correlation acceleration. All other functions should perform equally well in either case.

Additional Notes About Hypersignal Software Operation

1) Currently, external triggering operation is inactive with the TIGER 30 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

2) The BB71 analog I/O option differs from typical DSP board operation because of its 4-channel capability. Currently, channels 0 through 3 can be selected in any combination in the Analog Conversion function CHANNEL LIST field (e.g. '0,1,2,3', or '2,3') and any two can be selected simultaneously in the Digital Oscilloscope and Spectrum Analyzer functions (e.g. 'a1' and 'a3', or 'a0' and 'a3').

3.10 Innovation Integration

GENERAL NOTES

Before running DSPower and/or Hypersignal software with ADC64, M44 and PCxx boards, please read the following information.

1) Make sure your C:\config.sys file contains a line similar to 'DEVICE = C:\WINDOWS\EMM386.EXE', and make sure to add 'X=DC00-DFFF' to this line. This line should appear near the top of the config.sys file, immediately after the line similar to 'DEVICE = C:\WINDOWS\HIMEM.SYS' and before any other device drivers are loaded.

For example:

DEVICE = C:\WINDOWS\HIMEM.SYS

DEVICE = C:\WINDOWS\EMM386.EXE X = DC00-DFFF

DEVICE = ... other drivers ....

2) Make sure to disable the power.exe driver, if present in your config.sys file. If your config.sys file contains a line similar to 'device = C:\WINDOWS\POWER.EXE' or 'devicehigh = C:\WINDOWS\POWER.EXE', then add the word 'OFF'; e.g. 'devicehigh = C:\WINDOWS\POWER.EXE OFF'.

3) Make sure to reboot system to cause changes to take effect.

4) In DSPower software, be sure to enter correct clock rate value in Processor Clock field in the DSP/Analog I/O Hardware Manager dialog box.

For processor clock values other than default 40 MHz, enter the clock value also (e.g. 'IIC31-33', 'IIC31-50').

3.10.1 Innovative Integration ADC64 (TMS320C32)

Before the installation of Innovative Integration ADC64 board, please read the following information.

Please make sure the following factory default jumpers are not altered:

-jumper header JP28: 1-2, 3-4, 5-6 and 7-8 must all be installed (in)

When running Hypersignal software in user mode with the ADC64, use 'IIADC64' in System Configuration menu, and 'IIADC64-B' if you are using 4-channel option. For processor clock values other than default 40 MHz, enter the clock value also (e.g. 'IIADC64-60', 'IIADC64-B-60').

In DSPower software, be sure to enter correct clock rate value in Processor Clock field in the DSP/Analog I/O Hardware Manager dialog box. Entry is in MHz; for example '60' for 60 MHz board.

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

For more information about Innovative Integration ADC64, please refers to ADC64 hardware manual.

3.10.2 Innovative Integration M44 (TMS320C44)

Before the installation of Innovative Integration M44 board, please read the following information.

Factory default jumpers and switch settings which must be matched:

For M44 Revison D and earlier:

-jumper header JP13: 4-6, 8-10, 12-14 and 15-17 must be installed (in)

For M44 Revison E and later:

-jumper header JP13: 4-6, 8-10, 15-17 and 18-20 must be installed (in)

When running Hypersignal software in user mode with the M44, use 'IIM44' in System Configuration menu, and 'IIM44-MOD0-MOD1' if you are using analog I/O modules, where MOD0 stands for the module located at MODULE0 site and MOD1 stands for the module located at MODULE1 site. For example:

IIM44-A4D4 (one A4D4 located at MODULE0, only channels 0 - 3 are valid)

IIM44-NONE-A4D4 (one A4D4 located at MODULE1, only channels 4 - 7 are valid)

IIM44-A4D4-A4D4 (two A4D4s located at MODULE0 and MODULE1, channels from 0 - 7 are valid)

IIM44-AIX-DAC40 (one AIX located at MODULE0 and one DAC40 located at MODULE1)

For more information about Innovative Integration M44, please refers to M44 hardware manual.

3.10.3 Innovative Integration PC31 (TMS320C31)

Before the installation of Innovative Integration PC 31 board, please read the following information.

Factory default jumpers and switch settings which must be changed:

-jumper header JP17: for Rev. F and earlier, 9-10 must be removed (out), and 11-12 installed (in); for Rev. G and later, no changes needed

-jumper header JPADC: 7-8 must be out, and 1-2 must be in

-jumper header JP50: 7-8 must be out, and 1-2 must be in; 15-16 must be out, and 11-12 must be in
NOTE: pin 1 on JP50 is at left, not at right

-jumper header JP77: for Rev G. and later only, must be out

-jumper header JP66: for Rev G. and later only, must all be in (and JP65 jumpers must all be out)

-S2: If dual-port SRAM is installed, then S2 should be set for DC00H memory base address. For Rev. F and earlier, switch 8 must be on (for Rev. G and later, dual-port enable/disable is under software control). See also dual-port SRAM notes in file "addendum.txt" located in \HSMACRO directory.

Note that the above jumper changes essentially devote the 82C54 TMR0 counter to providing a periodic conversion clock for both ADC channels and first two DAC channels, and devote the 82C54 TMR1 counter to the second two DAC channels. This dedication of PC31 resources is necessary in order to maintain programmable, minimum-jitter sampling rates.

When running Hypersignal software in user mode with the PC31, use 'IIC31' in System Configuration menu for PC31 with dual-port SRAM and 'IIC31-B' for PC31 without dual-port SRAM (or with dual-port SRAM disabled). For processor clock values other than default 40 MHz, enter the clock value also (e.g. 'IIC31-33', 'IIC31-50').

In DSPower software, be sure to enter correct clock rate value in Processor Clock field in the DSP/Analog I/O Hardware Manager dialog box. Entry is in MHz; for example '60' for 60 MHz board.

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

Basic Descriptive and Installation Details

Use a hardware designator of 'IIC31' in the Hypersignal software System Configuration menu, and 'IIC31-B' for a board with no dual-port RAM installed. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The PC31 board contains the Texas Instruments TMS320C31 32-bit floating-point DSP, two analog input channels, which can also be multiplexed to provide up to 8 analog inputs, four analog output channels, a wide variety of external I/O connections, and onboard resources such as boot EPROM, RS-232, and programmable counter/timers. It is a full-length board with 16-bit ISA bus AT-style connector.

The default factory configuration PC31 board contains dual Burr-Brown DSP102 A/D converters and dual 4:1 analog input signal multiplexors, which can be programmed from Hypersignal for different sampling rates and acquisition using any combination of the 8 input channels. The DSP102 converters each provide one channel of 16-bit A/D conversion, and are both connected to TMS320C31 serial port 0, forming a stream of 32-bit serial words containing both converter results. The DSP102 converters are rated up to 200 kHz each, although standard Hypersignal instrument functions cannot achieve this rate without modification of TMS320C31 source code to eliminate some functionality, and continuous disk acquisition at rates above 100 kHz (one-channel) is heavily host PC-dependent. An 82C54 timer device is programmed by Hypersignal (timer 0) to provide software-selectable sampling rates to each DSP102 in the range of 150 Hz to 200 kHz. The recommended Hypersignal-compatible jumper configuration (see below) sets up both DSP102s to receive the same sampling clock, resulting in synchronous sampling in cases where each channel originates from separate DSP102 converters. From Hypersignal software menus, if more than two channels multiplexed to the same A/D converter are selected, then sampling is not synchronous. See notes under "Additional Notes About Hypersignal Software Operation" below for examples and detailed information about multichannel input operation.

The default factory configuration PC31 board contains dual Burr-Brown DSP202 stereo D/A converters. The DSP202 converters each provide two channels of 16-bit D/A conversion, and are both connected to TMS320C31 serial port 0 using a time-multiplexed method. Hypersignal allows output and signal generation using any combination of the four converters. The DSP202 converters are rated up to 200 kHz each, although standard Hypersignal instrument functions cannot achieve this rate without modification of TMS320C31 source code to eliminate some functionality, and continuous disk generation at rates above 100 kHz (one channel) is heavily host PC-dependent. An 82C54 timer device is programmed by Hypersignal (timers 0 and 1) to provide software-selectable sampling rates to each DSP202 in the range of 150 Hz to 200 kHz. The recommended Hypersignal-compatible jumper configuration (see below) sets up both DSP202s to receive the same sampling clock, but 180 degrees out of phase with respect to each other. This results in synchronous sampling on output channels 0 and 1, and on output channels 2 and 3, but not on all four output channels together. Note also that in the Hypersignal implementation, output channels 0 and 1 share the same sampling clock (and phase) as the first DSP102 A/D converter. See notes under "Additional Notes About Hypersignal Software Operation" below for examples and detailed information about multichannel input operation.

Hypersignal uses a default I/O base address of 280H and a default dual-port memory base address of DC00H for the PC31 board. Note that you *cannot* change these address from the System Configuration menu, as is the case with other boards. To change the base addresses, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal. See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the PC31 board I/O and memory base addresses in the "HSMACRO.CNF" (or "HSACOUS.CNF") file.

NOTE: Dual-port RAM is not required for PC31 use with Hypersignal. However, it is strongly recommended because it provides significantly better performance, especially in the areas of continuous signal acquisition/generation to/from hard disk and stimulus & response. If dual-port RAM is in use, we suggest you not change the default value of DC00H for the dual-port memory base address. This value will allow the board to work with network cards, "SuperVGA" video cards, and various resident software utilities and programs requiring extra "high memory". This value also preserves E000H-EFFFH as a possible EMS memory segment. Hypersignal software (and the board, if obtained through Signalogic) always defaults to the DC00H value.

Also, if dual-port RAM is used, be sure that you "block out" the region from DC00-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the PC31.

An entry of '-NN' after the entry for this board in the System Configuration menu will force Hypersignal sampling rate programming to be based on the value NN. NN should be specified in MHz; no entry sets the default value of 40 MHz, which is the value that is used by Signalogic in all PC31 systems obtained through Signalogic. But, the baseline model offered by Innovative Integration has a 33 MHz clock; in this case you should enter '-33.3333' after the System Configuration designators to inform the software about the slower speed. There is also a 50 MHz version of the board available. Example System Configuration menu entries might be 'IIC31-33.3333', or 'IIC31-B-50'.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

The PC31 board has an impressive variety of onboard resources and connections to the outside world. In fact, it bristles with extra features. Included are:

1) 48 bits of external, parallel digital I/O, in the form of two (2) 82C55 PIO devices

2) three (3) onboard counter/timers provided by an 82C54 device (in addition to the timers available inside the TMS320C31)

3) a DSPLink external bus connector

4) two (2) RS-232 serial ports in the form of an 85C30 dual UART device

5) boot EPROM option

6) great flexibility in jumpering interrupt sources, timer connections, and analog I/O sample clock sources for a wide variety of applications

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C3x code.

When using the PC31 board with Hypersignal software, the following jumper changes must be made from the initial factory defaults. PC31 boards obtained through Signalogic are delivered already configured in this manner:

1) Jumper header JP17: 9-10 must be removed (out), and 11-12 installed (in)

2) Jumper header JPADC: 7-8 must be out, and 1-2 must be in

3) Jumper header JP50: 7-8 must be out, and 1-2 must be in; 15-16 must be out, and 11-12 must be in

4) Switch S2: If dual-port SRAM is installed, then S2 should be set for DC00H memory base address, and switch 8 must be on. See additional dual-port SRAM notes below.

Note that the above jumper changes essentially dedicate the 82C54 TMR0 timer (counter 0) to providing a periodic conversion clock for both ADCs and for DAC0 and DAC1 (first dual DSP202 D/A converter), and dedicate the 82C54 TMR1 timer (counter 1) to DAC2 and DAC3 (second DSP202 D/A converter). The 82C54 counters are used in order to maintain programmable, linear, minimum-jitter sampling rates. This configuration leaves 82C54 TMR2 (counter 2) available to provide a system clock on PC31 external interrupt 3; alternatively, one of the two internal TMS320C31 timers can be used for this purpose. If the second two D/A converters are not needed, TMR1 can be re-routed for other uses, but in this case Hypersignal operations specifying output channels 2 and 3 may produce undefined results.

Dual-Port SRAM NOTES: **1) With Rev. F PC31 boards, we have found that dual-port SRAM may conflict with extended memory (above 1024 kbyte) on some clone machines at boot time. If a boot problem occurs with PC31 dual-port memory enabled, temporarily disable the dual-port memory (switch 8 on S2 to off), boot the machine, and enable the memory (switch 8 on). Yes, we know this means you must operate your system without the cover. Contact both Signalogic and Innovative Integration with the clone machine type, speed, and manufacturer.**

2) When using dual-port SRAM on the PC31, make sure to insert 'X=DC00-DFFF NOEMS' as the first attribute in the 'device = \windows\EMM386.EXE' line inside your config.sys file. See section 2.1 above, Compatibility Considerations, for more information.

3) If no dual-port SRAM is installed on your PC31, use the 'IIC31-B' selection in the System Config menu. Note, however, that this may result in as much as a 50% decrease in PC31 performance for some Hypersignal functions.

Hardware Ratings and Software Throughput Specifications

NOTE:

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C31, 50 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	85 kHz
1 channel Hard Disk Playback:	85 kHz
Max. 2-chn. Spectrum Analyzer Rate:	120 kHz
1024 pt FFT Accel. Disk File Rate:	20.3 frames/sec
Spectrum Analyzer Noise Floor:	85 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	32k x 32
Maximum Ext. Mem. Config:	512k x 32
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	yes (RS-232)
External Parallel Interface:	yes (DSPLink, Exp. bus)
Host Data Transfer Method:	Prog I/O and/or Memory-Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bit (8 multiplexed)
Codec channels:	none
Expandable:	yes, 8-chan multiplex
Simultaneous S/H	yes (2-chan only)
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±10V; RP
Input Impedance:	100k
Anti-Aliasing Filters:	no (Rev G: Butter4)
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152 Hz (40 MHz proc)
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	4, 16-bit
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±10V, ±1V, RP
Output Impedance:	UA
Reconstruction Filters:	no (Rev G: Butter 4)
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152 Hz (40 MHz proc)
External TTL-Level Trigger:	no

Onboard Memory Usage

The default factory memory configuration PC31 is 32k x 32 of onboard, external zero wait-state SRAM. Hypersignal uses less than 8k x 32 of this memory for program storage, and the remainder for analog data buffers and processing areas. A memory upgrade to 128k x 32 of onboard SRAM greatly improves stimulus & response operation by providing for larger stimuli and enables convolution/correlation acceleration; the 128k x 32 amount is strongly recommended. A memory upgrade to 512k x 32 is also available, but this memory must run at 1 wait-state at processor speeds above 33 MHz (note that 1 wait-state = 1 instruction cycle on 'C3x and 'C4x devices). The default factory configuration PC31 does not include the 2k x 32 dual-port RAM option, although Hypersignal drivers are optimized to make use of this option if installed. The dual-port option is strongly recommended as it can improve the performance of continuous disk transfer and other functions by as much as a factor of 2x. See the notes above for more information about PC31 dual-port memory.

Additional Notes About Hypersignal Software Operation

1) Currently, external triggering operation is inactive with the PC31 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

2) The multichannel input capability of the PC31 is a hybrid between standard two-channel boards, which typically provide synchronous sampling between the two channels, and "channel scan" boards, which provide evenly spaced sampling between the number of channels in use. A good example of the latter is the Data Translation DT282x series, which can multiplex up to 16 channels into one A/D converter, so that each channel is sampled at the overall rate divided by the number of channels on the scan list. The PC31 differs in that there are *two* multiplexors, one in front of each A/D converter. Below are some examples of what happens with various Hypersignal multichannel menu entries, and comments about the results. Note that examples below apply to *input* only:

CHANNEL LIST entry	Results / Comments
'0,4'	synchronous sampling at the entered rate on channels 0 and 4
'5,2'	synchronous sampling at the entered rate on channels 5 and 2; channel 5 is stored first
'0,1'	quadrature sampling at the entered rate on channels 0 and 1 (software uses an internal rate equal to 2x entered rate)
'0,1,2,3'	"octature" sampling at the entered rate on channels 0, 1, 2, and 3 (software uses an internal rate equal to 4x entered rate)
'0,1,4,5'	synchronous sampling at the entered rate on channels 0 and 4, synchronous sampling (but quadrature with respect to channels 0 and 4) sampling at the entered rate on channels 1 and 5 (software uses an internal rate equal to 2x entered rate)

3) It should be clear that with PC31 input, the so-called "synchronous pairs" are any pair of channels where one channel is 0-3 and the other is 4-7. This is because of the physical connection of channels 0-3 to the first multiplexor and channels 4-7 to the second multiplexor. Note also that any time the multiplexors must be "active" (changed dynamically during acquisition), the internal sampling rate becomes N times the entered rate, where N is the maximum number of channels selected on either multiplexor. This has two important ramifications:

- Much greater processing time requirements are imposed on the TMS320Cx, which can affect functions like continuous data transfer to/from hard disk. In fact, there are two versions of interrupt service routines used for PC31 input in Hypersignal TMS320C31 drivers: an optimized routine for any two-channel, non-multiplexed combination (synchronous pairs noted above), and a much slower routine for any multiplexed combination. Measurements at Signalogic using a 33 MHz 80386

machine have shown that continuous disk acquisition of channels 0-3 (four channels on same multiplexor) had a maximum rate of about 20 kHz per channel (aggregate rate of about 80 kHz), while acquiring only channel 0 had a maximum rate of more than 105,000 kHz. This example illustrates the difference in performance to be expected when the multiplexing code inside the software driver is in use.

- If the number of channels selected on each multiplexor is not the same, then N-M channel samples will be discarded on the multiplexor with fewer (M) channels selected, wasting valuable TMS320C3x processing time. In the example above, when acquiring channels 0-3, channels 4-7 are actually being transmitted over the serial port to the TMS320C31 also, even though they are not being used.

4) Since output on the PC31 does not use the same multiplexor-based scheme as input, there are some important notes to keep in mind when using PC31 output with Hypersignal:

- Output channel entry higher than 3 “wraps around”, starting at 1. An output channel entry of 4 will appear on 1, 5 will appear on 2, 6 on 3, and 7 on 0. This is because there are only four (4) D/A channels (DACs) on the PC31, while there are eight (8) A/D channels (when multiplexing is used). This scheme allows channel entries such as ‘0,4’ or ‘1,5’ to exhibit optimized performance for both input and output (see notes above about synchronous input channel pairs).
- DAC 2 and DAC 3 (third and fourth D/A converters) are *not* available during stimulus and response operation. Input can be looped back on DACs 2 and 3 in monitor mode, but no stimulus data will be output.
- When using “monitor mode” in either acquisition operation or stimulus and response measurement, multichannel entry should be made keeping in mind that entering more than four (4) channels may cause undefined results on the outputs.
- When in use, DACs 2 and 3 are about 180 degrees out of phase with respect to DACs 0 and 1. Thus, simultaneous sampling is occuring on DACs 0 and 1, and on DACs 2 and 3, but not on all four together.
- When all four input channels are in use, and all four DAC channels are in use, the DAC channels are actually being sampled twice as fast as necessary. However, since performance is limited in this mode, the maximum sampling rates will not be close to exceeding DAC maximum specifications.

For more information about Innovative Integration PC31, please refers to PC31 hardware manual.

3.10.4 Innovative Integration PC32 (TMS320C32)

Before the installation of Innovative Integration PC 32 board, please read the following information.

Currently, no factory default jumpers must be changed. The dual-port SRAM address (switch S1) should be changed to DC00H, to match the default Hypersignal value (this value is more compatible with typical Windows installations that include a network card). See also dual-port SRAM notes in file “addendum.txt” located in \HSMACRO directory, including directions on how to change the base address value.

NOTE ABOUT TWO-CHANNEL VERSION

When using a two-channel version of the PC32, external connector (DB37 connector on backplane) inputs B and D correspond to Hypersignal and DSPower software channels 0 and 1. For example, if channel ‘a0’ is entered into a Spectrum Analyzer field, then signal should be connected to input B on the backplane.

For correct analog I/O operation with Hypersignal and DSPower software, ensure that “ADC Trig” jumper JP5 is set to positions 1-2. Also, ensure that “DAC0", ”DAC1", “DAC2", and ”DAC3" jumpers JP10-14 are set to positions 1-2. This will cause both ADC and DAC sampling rate clocks to be generated by C3x onchip Timer0. This will provide simultaneous sampling on all analog I/O channels, which is the default mode.

When running Hypersignal software in user mode with the PC32, use ‘IIC32’ in System Configuration menu, and ‘IIC32-B’ for PC32 with only two (2) analog I/O channels installed. For processor clock values other than default 40 MHz, enter the clock value also (e.g. ‘IIC32-33’, ‘IIC32-50’, ‘IIC32-60’, IIC32-B-50’, etc.).

In DSPower software, be sure to enter correct clock rate value in Processor Clock field in the DSP/Analog I/O Hardware Manager dialog box. Entry is in MHz; for example ‘60’ for 60 MHz board.

For more information about Hypersignal software multichannel board operation, please see the “addendum.txt” file located in \HSMACRO directory.

For more information about Innovative Integration PC32, please refers to PC32 hardware manual.

3.10.5 Innovative Integration PC44 (TMS320C44)

Before the installation of Innovative Integration PC 44 board, please read the following information.

NOTE ABOUT ANALOG I/O

Factory default jumpers and switch settings which must be changed:

-jumper header JP15: 1-2 must be installed (in)

-jumper header JP37: 3-4, 9-10, 11-12 must be removed (out), and 5-6in

After making these changes, JP38 must only have 3-5, and 6-8 in

-jumper header JP38: 2-4, 10-12 must be out, and 3-5, 6-8 must be in.

After making these changes, JP38 must only have 3-5, and 6-8 in

-jumper header JP39: only 3-5 must be in

-jumper header JP43: 2-4 must be in for EINT0, 14-16 must be in for EINT2

-jumper header JP45, 46, 47, 48: only 2-3 must be in

The dual-port SRAM address (switch S2) should be changed to DC00H, to match the default Hypersignal value (this value is more compatible with typical Windows installations that include a network card). See also dual-port SRAM notes in file “addendum.txt” located in \HSMACRO directory, including directions on how to change the base address value.

When running Hypersignal software in user mode with the PC44, use ‘IIC44’ in System Configuration menu, and ‘IIC44-B’ if you wish analog I/O and other data buffers to be located in global SRAM. For processor clock values other than default 40 MHz, enter the clock value also (e.g. ‘IIC44-50’, ‘IIC44-B-50’).

In DSPower software, be sure to enter correct clock rate value in Processor Clock field in the DSP/Analog I/O Hardware Manager dialog box. Entry is in MHz; for example ‘60’ for 60 MHz board.

For more information about Hypersignal software multichannel board operation, please see the “addendum.txt” file located in \HSMACRO directory.

For more information about Innovative Integration PC44, please refers to PC44 hardware manual.

3.10.6 Innovative Integration PC50 (TMS320C50 or TMS320C51)

Factory default jumpers and switch settings which must be changed:

-jumper header JPADC1: 7-8 must be removed (out), and 1-1 installed (in)

-jumper header JPDAC: 7-8 must be out, and 1-2 must be in

-jumper header JP4: 1-2 must be in

-S2: If dual-port SRAM is installed, then S2 should be set for DC00H memory base address. See dual-port SRAM notes in file “addendum.txt” located in \HSMACRO directory.

Note that the above jumper changes essentially devote the 82C54 TMR1 counter to providing a periodic ADC and DAC conversion clock, in order to maintain programmable, minimum-jitter sampling rates.

When running Hypersignal software in user mode with the PC50, use ‘IIC50’ or ‘IIC51’ in System Configuration menu for PC50 with dual-port SRAM and ‘IIC50-B’ or ‘IIC51-B’ for PC51 without dual-port SRAM (or with dual-port SRAM disabled).

3.10.6 Innovative Integration PC50 (TMS320C50 or TMS320C51)

For processor clock values other than default 20 MHz, enter the clock value also (e.g. 'IIC50-33', 'IIC50-50', 'IIC50-60', 'IIC50-B-50', etc.).

In DSPower software, be sure to enter correct clock rate value in Processor Clock field in the DSP/Analog I/O Hardware Manager dialog box.

'IIC50-28.5' (TMS320C50, 57 MHz processor)

'IIC51-B-40' (TMS320C51, 80 MHz processor, no dual-port RAM or disabled)

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

For more information about Innovative Integration PC50, please refers to PC50 hardware manual.

Basic Descriptive and Installation Details

Use a hardware designator of 'IIC50' or 'IIC51' in the Hypersignal software System Configuration menu, and 'IIC50-B' or 'IIC51-B' for a board with no dual-port RAM installed. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The PC50 board contains either the Texas Instruments TMS320C50 or TMS320C51 16-bit fixed-point DSP, two analog input channels, which can also be multiplexed to provide up to 8 analog inputs, two analog output channels, a wide variety of external I/O connections, and onboard resources such as boot EPROM, and programmable counter/timers. It is a full-length board with 16-bit ISA bus AT-style connector.

The default factory configuration PC50 board contains one Burr-Brown stereo DSP102 A/D converter and dual 4:1 analog input signal multiplexors, which can be programmed from Hypersignal for different sampling rates and acquisition using any combination of the 8 input channels. The DSP102 provides two channels of 16-bit A/D conversion; one channel is connected to the TMS320C5x serial port, and the other is connected to the TMS320C5x TDM port, allowing both channels to be read during each sample interrupt. The DSP102 channels are rated up to 200 kHz each, although standard Hypersignal instrument functions cannot achieve this rate without modification of TMS320C5x source code to eliminate some functionality, and continuous disk acquisition at rates above 100 kHz (one-channel) is heavily host PC-dependent. An 82C54 timer device is programmed by Hypersignal (timer 0) to provide software-selectable sampling rates to the DSP102 in the range of 150 Hz to 200 kHz. The recommended Hypersignal-compatible jumper configuration (see below) sets up both DSP102 channels to receive the same sampling clock, resulting in synchronous sampling in cases where each channels are not multiplexed (see comments below). From Hypersignal software menus, if more than two channels multiplexed to the same A/D converter are selected, then sampling is not synchronous. See notes under "Additional Notes About Hypersignal Software Operation" below for examples and detailed information about multichannel input operation.

The default factory configuration PC50 board contains one Burr-Brown DSP202 stereo D/A converter. The DSP202 converter provides two channels of 16-bit D/A conversion; one channel is connected to the TMS320C5x serial port, and the other to the TMS320C5x TDM port. Hypersignal allows output and signal generation using any combination of the two channels. The DSP202 channels are rated up to 200 kHz each, although standard Hypersignal instrument functions cannot achieve this rate without modification of TMS320C31 source code to eliminate some functionality, and continuous disk generation at rates above 100 kHz (one channel) is heavily host PC-dependent. An 82C54 timer device is programmed by Hypersignal (timer 0) to provide software-selectable sampling rates to each DSP202 in the range of 150 Hz to 200 kHz. The recommended Hypersignal-compatible jumper configuration (see below) sets up both DSP202 channels to receive the same sampling clock. Note also that in the Hypersignal implementation, output channels 0 and 1 share the same sampling clock (and phase) as the two DSP102 A/D channels. See notes under "Additional Notes About Hypersignal Software Operation" below for examples and detailed information about multichannel input operation.

Hypersignal uses a default I/O base address of 280H and a default dual-port memory base address of DC00H for the PC50 board. Note that you *cannot* change these address from the System Configuration menu, as is the case with other boards. To change the base addresses, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal. See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the PC50 board I/O and memory base addresses in the "HSMACRO.CNF" (or "HSACOUS.CNF") file.

NOTE: Dual-port RAM is not required for PC50 use with Hypersignal. However, it is recommended because it provides better performance, especially in the areas of continuous signal acquisition/generation to/from hard disk and stimulus & response. If dual-port RAM is in use, we suggest you not change the default value of DC00H for the dual-port memory base address. This value will allow the board to work with network cards, "SuperVGA" video cards, and various resident software utilities and programs requiring extra "high memory". This value also preserves E000H-EFFFH as a possible EMS memory segment. Hypersignal software (and the board, if obtained through Signalogic) always defaults to the DC00H value.

Also, if dual-port RAM is used, be sure you "block out" the region from DC00-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the PC50.

An entry of '-NN' after the entry for this board in the System Configuration menu will force Hypersignal sampling rate programming to be based on the value NN. NN should be specified in MHz; no entry specifies the default value of 20 MHz, which is the default factory clock input. Note that the actual processor speed is twice the clock input for the PC50 board. Thus, a '-20' entry would apply for a 40 MHz processor. There are also 57 and 80 MHz versions of the processor available; in these cases, entry should be '-28.5' and '-40', respectively. Example System Configuration menu entries might be 'IIC50-28.5', or 'IIC51-B-80'.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

The PC50 board has an impressive variety of onboard resources and connections to the outside world. Included are:

1) 28 bits of external, parallel digital I/O, in the form of two (2) 82C55 PIO devices

2) three (3) onboard counter/timers provided by an 82C54 device (in addition to the timers available inside the TMS320C31)

3) a DSPLink external bus connector

4) boot EPROM option

5) great flexibility in jumpering interrupt sources, timer connections, and analog I/O sample clock sources for a wide variety of applications

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C5x code.

When using the PC50 board with Hypersignal software, the following jumper changes must be made from the initial factory defaults. PC50 boards obtained through Signalogic are delivered already configured in this manner:

1) Jumper header JPADC1: 7-8 must be removed (out), and 1-1 installed (in)

2) Jumper header JPDAC: 7-8 must be out, and 1-2 must be in

3) Jumper header JP4: 1-2 must be in

4) Switch S2: If dual-port SRAM is installed, then S2 should be set for DC00H memory base address. See dual-port SRAM notes below.

Note that the above jumper changes essentially dedicate the 82C54 TMR0 timer (counter 0) to providing a periodic conversion clock for both ADCs and for the dual DSP202 D/A converter. An 82C54 counter is used in order to maintain programmable, linear, minimum-jitter sampling rates. This configuration leaves 82C54 TMR1 and TMR2 (counters 1 and 2) available to provide user-defined timebases such as a system clock.

Dual-Port SRAM NOTES:

1) When using dual-port SRAM on the PC50, make sure to insert 'X=DC00-DFFF NOEMS' as the first attribute in the 'device = \windows\EMM386.EXE' line inside your config.sys file. See section 2.1 above, Compatibility Considerations, for more information.

2) If no dual-port SRAM is installed on your PC50, use the 'IIC50-B' or 'IIC51-B' selections in the System Config menu.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C50 or TMS320C51, 50 nsec
Data Format, Wordlength:	fixed-point, 16-bit
1 channel Hard Disk Record:	85 kHz
1 channel Hard Disk Playback:	85 kHz
Max. 2-chn. Spectrum Analyzer Rate:	120 kHz
1024 pt FFT Accel. Disk File Rate:	16.3 frames/sec
Spectrum Analyzer Noise Floor:	85 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	64k x 16 data, 64k x 16 prog
Maximum Ext. Mem. Config:	64k x 16 data, 64k x 16 prog
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes (DSPLink, Exp. bus)
Host Data Transfer Method:	Prog I/O and/or Memory-Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bit (8 multiplexed)
Codec channels:	none
Expandable:	yes, 8-chan multiplex
Simultaneous S/H	yes (2-chan only)
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±10V; RP
Input Impedance:	100k
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152 Hz (40 MHz proc)
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bit
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±10V, ±1V, RP
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152 Hz (40 MHz proc)
External TTL-Level Trigger:	no

Onboard Memory Usage

The default factory memory configuration PC50 is 64k x 16 of onboard, external zero wait-state SRAM for both program and data areas. Hypersignal uses less than 8k x 32 of the program memory for code storage, and all of the data memory area for ana-

log data buffers and processing areas. The standard amount of memory is the maximum; this amount provides for reasonable stimulus & response operation and 2-channel spectrum analyzer operation, and enables convolution/correlation acceleration. The default factory configuration PC50 does not include the 2k x 32 dual-port RAM option, although Hypersignal drivers are optimized to make use of this option if installed. The dual-port option is recommended as it can improve the performance of continuous disk transfer and other functions. See the notes above for more information about PC50 dual-port memory.

Additional Notes About Hypersignal Software Operation

1) Currently, external triggering operation is inactive with the PC50 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

2) The multichannel input capability of the PC50 is a hybrid between standard two-channel boards, which typically provide synchronous sampling between the two channels, and "channel scan" boards, which provide evenly spaced sampling between the number of channels in use. A good example of the latter is the Data Translation DT282x series, which can multiplex up to 16 channels into one A/D converter, so that each channel is sampled at the overall rate divided by the number of channels on the scan list. The PC50 differs in that there are *two* multiplexors, one in front of each A/D converter.

Currently, Hypersignal software treats the PC50 board as only a two-channel board. For full 8-channel operation, the user is advised to modify the TMS320C5x source code, using the PC31 board from Innovative Integration as a guide. The analog I/O on this board is very similar to the PC50, and the Hypersignal TMS320C3x source code fully implements every feature for the PC31 board. Below are some examples of what happens with various Hypersignal multichannel menu entries, and comments about the results. Note that examples below apply to *input* only:

CHANNEL LIST entry	Results / Comments
'0,4'	synchronous sampling at the entered rate on channels 0 and 4
'5,2'	synchronous sampling at the entered rate on channels 5 and 2; channel 5 is stored first

3) It should be clear that with PC50 input, the so-called "synchronous pairs" are any pair of channels where one channel is 0-3 and the other is 4-7. This is because of the physical connection of channels 0-3 to the first multiplexor and channels 4-7 to the second multiplexor. Currently, specifying channel entry such that more than one channel is in the same "multiplexor half" will produce undefined results.

3.10.7 Innovative Integration Modules

A. NOTE ABOUT A4D4 MODULE

Factory default jumpers and switch settings which must be matched:

-jumper header JP23: 1-2 must be installed (in).

Note that the above jumper change essentially devotes the DDS Synthsizer to provide an input clock to the 82C54 TMR1 counter.

When using Signalogic software, for A4D4 module seated in MODULE0, only channels 0 to 3 are valid; for A4D4 module seated in MODULE1, channels are numbered from 4 to 7. Entering invalid channel numbers in Hypersignal or DSPower software will cause undefined results. For example, entering channels 4 - 7 when an A4D4 module is installed only in MODULE0 site is undefined. If invalid channel numbers and/or invalid modules specified or not assigned to the correct site, then the board may need a "hard" reset in order to resume normal operation. For example, in a Hypersignal instrument display, if the display locks due to invalid entry, then it may be necessary to return to the main menu and re-enter the instrument. As another example, in a DSPower-HwLib program, it may be necessary to make a DSInitBoard() function call to resume normal operation.

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

B. NOTE ABOUT AIX MODULE

The AIX module uses IIOF0 and IIOF2. Please make sure that these interrupts are set as described in general M44 information.

Note that the default DSP executable program (COFF) file supplied by Signalogic does not allow an AIX module to operate together with an A4D4 on the same M44 board. The AIX uses the DDS Synthesizer directly for sampling clock while the A4D4 uses the DDS Synthsizer to provide an input clock to the 82C54 TMR1 counter. Using these two module at the same time will cause timing problems. To resolve this, the DSP program file should be modified to ensure correct timing to the corresponding modules.

When using Signalogic software, for AIX module seated in MODULE0, only channels 0 to 3 are valid; for AIX module seated in MODULE1, channels are numbered from 4 to 7. Entering invalid channel numbers in Hypersignal or DSPower software will cause undefined results. For example, entering channels 4 - 7 when an AIX module is installed only in MODULE0 site is undefined. If invalid channel numbers and/or invalid modules specified or not assigned to the correct site, then the board may need a "hard" reset in order to resume normal operation. For example, in a Hypersignal instrument display, if the display locks due to invalid entry, then it may be necessary to return to the main menu and re-enter the instrument. As another example, in a DSPower-HwLib program, it may be necessary to make a DSInitBoard() function call to resume normal operation.

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

C. NOTE ABOUT AIX20 MODULE

The AIX20 module uses IIOF0 and IIOF2. Please make sure that these interrupts are set as described in general M44 information.

Note that the default DSP executable program (COFF) file supplied by Signalogic does not allow an AIX20 module to operate together with an A4D4 on the same M44 board. The AIX uses the DDS Synthesizer directly for sampling clock while the A4D4 uses the DDS Synthsizer to provide an input clock to the 82C54 TMR1 counter. Using these two module at the same time will cause timing problems. To resolve this, the DSP program file should be modified to ensure correct timing to the corresponding modules.

When using Signalogic software, for AIX20 module seated in MODULE0, only channels 0 to 3 are valid; for AIX module seated in MODULE1, channels are numbered from 4 to 7. Entering invalid channel numbers in Hypersignal or DSPower software will cause undefined results. For example, entering channels 4 - 7 when an AIX20 module is installed only in MODULE0 site is undefined. If invalid channel numbers and/or invalid modules specified or not assigned to the correct site, then the board may need a "hard" reset in order to resume normal operation. For example, in a Hypersignal instrument display, if the display locks due to invalid entry, then it may be necessary to return to the main menu and re-enter the instrument. As another example, in a DSPower-HwLib program, it may be necessary to make a DSInitBoard() function call to resume normal operation.

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

D. NOTE ABOUT DAC40 MODULE

The DAC40 module uses IIOF0 and IIOF2. Please make sure that these interrupts are set as described in general M44 information.

Note that the default DSP executable program (COFF) file supplied by Signalogic does not allow different modules to operate together on the same M44 board. The DAC40 uses the DDS Synthesizer directly for sampling clock while a diferent module may use the DDS Synthsizer for other timing sources. Using two different modules simultanesously will cause timing problems. To resolve this, the DSP program file should be modified to ensure correct timing to the corresponding modules.

When using Signalogic software, for DAC40 module seated in MODULE0, only channels 0 to 3 are valid; for DAC40 module seated in MODULE1, channels are numbered from 4 to 7. Entering invalid channel numbers in Hypersignal or DSPower software will cause undefined results. For example, entering channels 4 - 7 when an DAC40 module is installed only in MODULE0 site is undefined. If invalid channel numbers and/or invalid modules specified or not assigned to the correct site, then the board may need a "hard" reset in order to resume normal operation. For example, in a Hypersignal instrument display, if the display locks due to invalid entry, then it may be necessary to return to the main menu and re-enter the instrument. As another example, in a DSPower-HwLib program, it may be necessary to make a DSInitBoard() function call to resume normal operation.

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

E. NOTE ABOUT SD4 MODULE

The SD4 module uses IIOF0 and IIOF2. Please make sure that these interrupts are set as described in general M44 information.

Note that the default DSP executable program (COFF) file supplied by Signalogic does not allow different modules to operate together on the same M44 board. The SD4 uses the DDS Synthesizer directly for sampling clock while a diferent module may use the DDS Synthsizer for other timing sources. Using two different modules simultanesously will cause timing problems. To resolve this, the DSP program file should be modified to ensure correct timing to the corresponding modules.

When using Signalogic software, for SD4 module seated in MODULE0, only channels 0 to 3 are valid; for SD4 module seated in MODULE1, channels are numbered from 4 to 7. Entering invalid channel numbers in Hypersignal or DSPower software will cause undefined results. For example, entering channels 4 - 7 when an SD4 module is installed only in MODULE0 site is undefined. If invalid channel numbers and/or invalid modules specified or not assigned to the correct site, then the board may need a "hard" reset in order to resume normal operation. For example, in a Hypersignal instrument display, if the display locks due to invalid entry, then it may be necessary to return to the main menu and re-enter the instrument. As another example, in a DSPower-HwLib program, it may be necessary to make a DSInitBoard() function call to resume normal operation.

For more information about Hypersignal software multichannel board operation, please see the "addendum.txt" file located in \HSMACRO directory.

F. NOTE ABOUT SD16 MODULE

Please refers to SD16 hardware manual.

3.11 Intelligent Instrumentation, Inc. (I^3)

Hypersignal supports both the PCI20202C "smart carrier" with the Texas Instruments TMS320C25 DSP chip and the PCI20041C-3A DMA carrier with no DSP chip. A series of standard analog I/O daughterboard "modules" can be connected to the carriers. Hypersignal performs automatic detection and operation of certain modules and combinations of modules.

3.11.1 I^3 PCI-20202C (TMS320C25)

Basic Descriptive and Installation Details

Use a hardware designator of 'PCI202C' for the PCI20202C-1 carrier (28 MHz processor speed), 'PCI202C-2' for the PCI20202C-2 carrier (40 MHz processor speed), and 'PCI202C-3' for the PCI20202C-3 carrier (50 MHz processor speed) in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The PCI20202C system consists of a "carrier" board with several combinations of analog I/O daughterboard "modules" that can be connected to the carrier board. The basic carrier interface provides a programmed I/O transfer interface. The analog I/O modules provide anything from 8-channel A/D to 32-channel A/D to 2-channel D/A, and at varying speeds. It is a full-length board with 8-bit ISA bus XT-style connector.

Carrier Configuration

For proper Hypersignal operation, the following jumpers should be checked and/or added to the carrier board from its factory configuration:

- Establish SYNC routing by inserting W2, W7, W11, and W12.
- You may be able to choose the "Fast I/O Timing" for modules 1 and 2, depending on which module configuration is being used. These jumpers are W25 and W26.
- No interrupts.
- No wait-states.

The carrier base address should be selected so that it does not interfere with other cards in your system. Hypersignal allows you to use any carrier-supported base address by changing the I/O BASE ADDRESS field in the System Configuration menu. The default I/O BASE ADDRESS setting is 300H, which is also the initial factory default. You must change this field at the value marked "BB" if you change the base address jumpers on the carrier.

3.11.1 I^3 PCI-20202C (TMS320C25)

Module Overview

Below are channel numbers and resolutions (bits/channel), sampling rates (given as maximum for *single* channel operation), and method of data transfer for the various acquisition modules (all analog channels are assumed to be single-ended unless otherwise noted):

PCI-20019M-1A	8-channel 12-bit linear, 89 kHz A/D, programmed I/O transfer
PCI-20023M-1	8-channel 12-bit linear, 180 kHz A/D, programmed I/O transfer
PCI-20031M-1	32-channel 12-bit linear, rate is 19M or 23M dependent, programmed I/O transfer
PCI-20003M-2	2-channel 12-bit linear, 80 kHz D/A, programmed I/O transfer

Using the High-Speed Acquisition Modules

The PCI-20019M-1A or PCI-20023M-1 A/D modules may be used in either carrier position 1 or 2. Each of these modules may be jumpered for automatic channel scan or software channel select; both of these jumper options are supported. Keep in mind the available Hypersignal functions associated with each choice:

software channel select: Analog Conversion (2-channel A/D only), 2-channel Digital Oscilloscope, and 2-channel Spectrum Analyzer (see below for jumper selections).

automatic channel scan: Analog Conversion (8-channel A/D only). Digital Oscilloscope and Spectrum Analyzer *are not* supported with this configuration (see below for jumper selections).

Analog channels for these modules are entered as '0' through '7' in the Hypersignal functions.

The high-speed acquisition modules are factory configured to automatic channel scan mode; to choose software channel select mode, jumper W11 must be IN, and jumpers W12 and W27 must be OUT. All other jumpers should be left at factory configuration (unless the analog expander/sequencer is being used, in which case the daisy chain inputs should be connected to the sample; see Using the Analog Expander/Sequencer Module, below.

NOTE: Keep in mind that the actual, overall sampling rate used by the PCI-20202C system is the desired (entered) rate multiplied by the number of channels in use. This means that even though you enter and see the sampling rate *per channel* when working with Hypersignal, the board may actually be going faster, and could even be getting close to its maximum rating (see Module Overview, above). This becomes more important when using automatic channel scan (Analog Conversion only), because the scan process must always scan the lowest channel entered in the CHANNEL LIST field up through 7. For example, a '0,4-6' entry would cause the full 8-channel scan (0 through 7), even though only channels 0, 4, 5, and 6 would be retained in the continuous input waveform file.

Using the Analog Expander/Sequencer Module

If the PCI-20031M-1 expander/sequencer module is used, it should be placed in carrier position 1, with a high-speed A/D acquisition module in position 2. The expander/sequencer allows multichannel acquisition of up to 32 single-ended channels. Expander/sequencer channels are represented in Hypersignal by channels 8-39; channels 0-7 always refer to the high-speed acquisition module. Sequential channels may be entered as the first channel, a dash, and the last channel. For example, '8-12,24' would acquire expander/sequencer channels 8 through 12 and also channel 24.

Hypersignal only supports the PCI-20031M-1 module in position 1, with a high-speed acquisition module simultaneously present in position 2. All jumpers except for channel advance should be set to factory configuration. Channel advance should be on "software command only": change factory configuration from W8 IN to W8 OUT, and change W16 OUT to W16 IN.

The PCI-20019M-1A or PCI-20023M-1 high-speed module (position 2) must have their daisy chain inputs connected to the sample/hold inputs; this corresponds to the jumpers W14, W16, W18, and W19 being changed from OUT (factory configuration) to IN.

NOTE: Keep in mind that the actual, overall sampling rate used by the PCI20202C system is the desired (entered) rate multiplied by the number of channels in use. This means that even though you enter and see the sampling rate *per channel* when working with Hypersignal, if you scan several channels with the expander/sequencer the board may actually be going faster, and getting close to or exceeding its maximum rating (see Module Overview, above).

Using the Analog Output Module

Used in conjunction with a high-speed acquisition module, the PCI-20003M-x will generate analog output from previously stored waveform files. Hypersignal uses this module to provide D/A output capability from the Analog Conversion functions, and also target section playback from time and frequency domain displays.

No jumper changes are needed from the PCI-20003M-x factory configuration.

Possible Module Configurations

Hypersignal allows a number of different module configuration options to be used with the carrier board. Invalid combinations are detected and warning messages displayed after the System Configuration function is executed.

Module Position 1	Module Position 2
PCI-20019M-1A *or* PCI-20023M-1	PCI-20003M-2 *or* *none installed*
PCI-20003M-2 *or* *none installed*	PCI-20019M-1A *or* PCI-20023M-1
PCI-20031M-1	PCI-20019M-1A *or* PCI-20023M-1

Digital triggering is supported in the Analog Conversion, Digital Oscilloscope, and Spectrum Analyzer functions. Through field entry and/or instrumentation display controls in these functions, positive or negative values are sent down to the TMS320C25 code and digitally compared against incoming samples. Multichannel triggering combinations (e.g. using one channel solely for data and the other(s) for data) is supported.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -2047 to 2047 for 12-bit A/D) and applied digitally, via TMS320C25 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C25, 125 nsec
Data Format, Wordlength:	fixed-point, 16-bit
1 channel Hard Disk Record:	70 kHz
1 channel Hard Disk Playback:	70 kHz
Max. 2-chn. Spectrum Analyzer Rate:	25 kHz
1024 pt FFT Accel. Disk File Rate:	7.9 frames/sec
Spectrum Analyzer Noise Floor:	67 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 16 P; 16k x 16 D
Maximum Ext. Mem. Config:	32k x 16 P; 64k x 16 D
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, Analog Expander/Sequencer
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	8, 12-bits (up to 32, 12-bits)
Codec channels:	none
Expandable:	yes, up to 40, 12-bits
Simultaneous S/H	no (optionally available)
Typical Dynamic Range:	67 dB
THD:	UA
Voltage Range:	±2.5V to ±10V; J3
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	89 kHz-1CH
Min. Sampling Rate:	.02 Hz
External TTL-Level Trigger:	yes
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 12-bits
Codec channels:	none
Expandable:	yes
Simultaneous S/H:	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±2.5V to ±10V; J3
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	89 kHz
Min. Sampling Rate:	.02 Hz
External TTL-Level Trigger:	yes

Onboard Memory Usage

The PCI20202C carrier is shipped standard with 16k x 16 of program memory and 16k x 16 of data memory. Hypersignal uses less than 8k x 16 of program memory. The Intelligent Instrumentation "memory expansion module" will expand the PCI20202C carrier memory to 32k x 16 of both program and data memory. With Hypersignal, this will significantly increase continuous disk transfer rates, increase 2-channel Spectrum Analyzer FFT size to 2048, enable convolution/correlation acceleration, and enable stimulus & response operations (using the extra 24k x 16 of program memory for stimulus storage).

Additional Notes About Hypersignal Software Operation

There are only minor deviations from standard Hypersignal operation for the PCI20202C system:

1) The external trigger option in the Analog Conversion menu is not active.

2) After

- entering 'PCI202C' for any or all of the ANALOG CONVERSION, DSP ACCELERATOR, DIGITAL SCOPE, and SPECTRUM ANALYZER fields, and
- selecting a I/O BASE ADDRESS (in the spot corresponding to "II", or Intelligent Instrumentation) value for the carrier (default is 300 hex), and
- exiting the System Configuration menu,

a full-screen PCI-20202C configuration display should appear similar to the screen shown in Figure 3-1, below. This screen is useful for making sure that the software properly sees the carrier. It is also allows you to check what modules are in your carrier. Pressing any key while in the configuration display will return you to the main menu display.

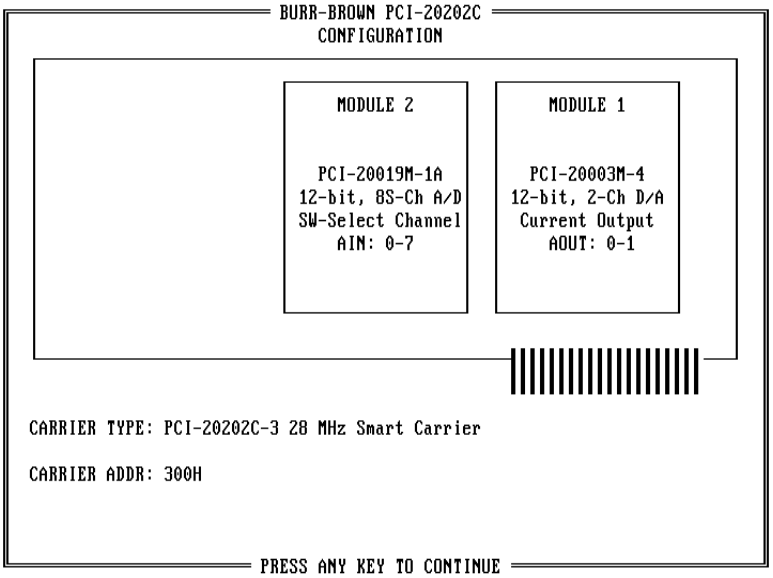

Figure 3-1

3.11.2 I^3 PCI-20041C (No DSP Chip)

NOTE: Even though this board has no DSP chip, Spectrum Analyzer and DSP Accelerator operation can still be achieved via a separate, inexpensive "DSP Coprocessor" board. See section 3.5.1, above, for a description of one such board. See the product catalog for pricing information and further details.

Basic Descriptive and Installation Details

Use a hardware designator of 'PCI41C' in the Hypersignal software System Configuration menu (except for DSP ACCELERATOR field). It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The PCI20041C system consists of a "carrier" board with several combinations of analog I/O daughterboard "modules" that can be connected to the carrier board. The basic carrier interface provides a single-channel host DMA interface. The analog I/O modules provide anything from 8-channel A/D to 32-channel A/D to 2-channel D/A, and at varying speeds. It is a full-length board with 8-bit ISA bus XT-style connector.

Carrier Configuration

Hypersignal supports the -3A carrier which provides the DMA capability. The -2A carrier is not supported. The -3A carrier must be configured for the following functions:

- No interrupts.
- No wait-states.
- Sync In/Sync Out system connects Pacer Clock to all modules.
- DMA channel 1.
- Master carrier operation.

The carrier memory base address should be selected so that it does not interfere with other cards in your system. Hypersignal allows you to use any carrier-supported base address by changing the MEM BASE ADDRESS field in the System Configuration menu at the spot marked "BB". Refer to the carrier Users Manual for further information. Initial default Hypersignal memory base address for the PCI20041C carrier is D000H. Usually a memory segment base address of D000H is recommended for

IBM-AT and compatible machines, and an address of E000H for IBM-PC and compatible machines. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS and MEM BASE ADDRESS fields, for details on changing these addresses.

NOTE: In many late-model systems, part or all of the D000H and E000H 64 kbyte memory segments are needed for a network card, "SuperVGA" video card, and/or EMS memory. In such a case, we suggest using the E000H segment for EMS memory, placing the video card in a mode that doesn't require extra memory, setting the network memory address to D000H, and setting the PCI20041C address to D200H or higher. Hopefully, this will allow all resources to coexist peacefully.

Also, be sure that you "block out" the region from D000-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the PCI20041C.

The carrier must have the following jumpers installed. Note that only jumpers that are to be installed are listed. All other jumpers should not be installed.

SYNC IN/SYNC OUT

W27	IN
W3	IN
W7	IN
W11	IN

DMA CHANNEL 1/ MASTER CARRIER

W38-W39	IN
W28	IN
W30	IN

Module Overview

Modules supported by Hypersignal include:

PCI-20003M-x	2-channel 12-bit analog output; x=1 for voltage ouput, x=2 for voltage and current output
PCI-20019M-1	8-channel 12-bit high-speed analog input
PCI-20023M-1	8-channel 12-bit very high-speed analog input
PCI-20031M-1	32-channel high-speed expander/sequencer

PCI-20003M-1 and PCI-20003M-2 Configuration

This D/A module may reside in any module position. The module must be configured for either ±5 or ±10 Volt bipolar outputs.

±5V OUTPUT

W2	IN
W3	IN
W8	IN
W9	IN
W13	IN

±10V OUTPUT

W2	IN
W8	IN
W13	IN

PCI-20019M-1 Configuration

This A/D module may reside in any module position. The module must be configured for the following functions:

- Bipolar voltage range (±5 or ±10 V).
- Start-Conversion is SYNC IN.
- SYNC OUT at end of conversion (EOC) LOW.
- Automatic selective channel scan (n-7).
- Daisy chain input to sample/hold input (when using 31M E/S).

The module must have the following jumpers installed. Note that only jumpers that are to be installed are listed. All other jumpers should not be installed.

±5V INPUT

W2	IN
W4	IN

±10V INPUT

W2	IN

W4 IN

SYNC IN START CONVERSION

W8 IN

W9 IN

SYNC OUT AT EOC

W30 IN

AUTO. SELECT. CH. SCAN

W12 IN

W27 IN

W22 IN

DAISY CHAIN INPUT (WITH E/S ONLY)

W14 IN

W16 IN

W18 IN

W19 IN

PCI-20023M-1 Configuration

This A/D module may reside in any module position. The module must be configured for the following functions:

- Bipolar voltage range (±5 or ±10 V).
- Start-Conversion is SYNC IN.
- SYNC OUT at end of conversion (EOC) LOW.
- Automatic selective channel scan (n-7).
- Daisy chain input to sample/hold input (when using 31M E/S).
- Clock Speed select (W34) if computer clock > 10 MHz.

The module must have the following jumpers installed. Note that only jumpers that are to be installed are listed. All other jumpers should not be installed.

±5V INPUT

W2	IN
W4	IN

±10V INPUT

W2	IN
W4	IN

SYNC IN START CONVERSION

W8	IN
W9	IN

SYNC OUT AT EOC

W30	IN

AUTO. SELECT. CH. SCAN

W12	IN
W27	IN

DAISY CHAIN INPUT (WITH E/S ONLY)

W14	IN
W16	IN
W18	IN
W19	IN

PCI-20031M-1 Configuration

Hypersignal supports one or two Expander/Sequencer (E/S) module(s) on the carrier. A carrier with one E/S will consist of 40 analog input channels where channels 0-7 correspond to the A/D module channels, and channels 8-39 correspond to the E/S channels. A carrier with two E/S modules will additionally support E/S channels 40-71. Note that the E/S module(s) must be in module positions that immediately precede the A/D module position. Therefore, valid configurations include:

- E/S module in pos 1 and A/D module in pos 2.
- E/S module in pos 2 and A/D module in pos 3.

E/S module in pos 1, E/S module in position 2, and A/D module in pos 3.

The module(s) must be configured for the following functions:

- Channel advance on + edge of SYNC IN
- CLK Cycles of delay is 9.

The module(s) must have the following jumpers installed. Note that only jumpers that are to be installed are listed. All other jumpers should not be installed.

CHANNEL ADV ON + EDGE OF SYNC IN

W8 IN

CLK CYCLES OF DELAY

W4 IN

Input and output offset entries in the Analog Conversion function are active (enter in range -2048 to 2047 for 12-bit A/D) and applied digitally, via 80x86 assembly language. However, use of non-zero offsets can greatly reduce the maximum rate of continuous transfer to/from disk. This results from the host 80x86 processor handling each point individually in order to add an offset, precluding a pure DMA transfer and slowing the overall input/output rate.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	None
Data Format, Wordlength:	None
1 channel Hard Disk Record:	110 kHz
1 channel Hard Disk Playback:	55 kHz
Max. 2-chn. Spectrum Analyzer Rate:	55 kHz
1024 pt FFT Accel. Disk File Rate:	N/A
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	N/A
Maximum Ext. Mem. Config:	N/A
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	DMA

A/D SPECIFICATIONS

Number of linear PCM channels:	8, 12-bits (up to 32, 12-bit)
Codec channels:	none
Expandable:	yes (up to 72, 12-bits)
Simultaneous S/H	no
Typical Dynamic Range:	67 dB
THD:	UA
Voltage Range:	±2.5V to ±10V; J3
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	180 kHz
Min. Sampling Rate:	.02 Hz
External TTL-Level Trigger:	yes
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 12-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	UA
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±2.5V to ±10V; J3
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	180 kHz
Min. Sampling Rate:	.02 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

There is no onboard PCI-20041C memory; however, I/O is memory mapped and is accessible.

Additional Notes About Hypersignal Software Operation

There are several significant deviations from standard Hypersignal operation for the PCI-20041C system:

1) After

- entering 'PCI41C' for any or all of the ANALOG CONVERSION, DIGITAL SCOPE, and SPECTRUM ANALYZER fields, and
- selecting a MEM BASE ADDRESS (in the spot corresponding to "II", or Intelligent Instrumentation) value for the carrier (default is D000 hex), and
- exiting the System Configuration menu,

a full-screen PCI-20041C configuration display should appear as shown in Figure 3-2, below. This screen is useful for making sure that the software properly sees the carrier. It is also allows you to check what modules are in your carrier. Pressing any key while in the configuration display will return you to the main menu display.

Figure 3-2

2) The Analog Conversion TRIGGER DELAY field should be entered in the form of 'nnn' (units are in µsec) and can be either negative or positive. A negative trigger would cause the hardware to retain the specified amount of data prior to the trigger event in order to provide a "pre-history".

3) The "snap-in" real-time digital filtering, stimulus & response, and "monitor mode" (input loopbacked to output) functions are disabled—these functions are only enabled hardware with an onboard DSP chip.

3.11.3 I^3 ZPB32 (DSP32)

NOTE: This card can also serve as a "DSP Coprocessor" for supported acquisition boards that have no DSP chip, allowing Spectrum Analyzer and DSP Accelerator operation with these "analog-only" boards.

Basic Descriptive and Installation Details

Use a hardware designator of 'ZPB32' in the Hypersignal software System Configuration menu for a ZPB32 board with no analog I/O interface boards connected, 'ZPB32-ZPB1100' for a ZPB1100 and/or ZPB2100 board connected, and 'ZPB32-ZPD1007' for a ZPD 1007 external unit connected. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or not.

The ZPB32 contains the 32-bit floating-point AT&T DSP32 DSP chip, and a serial interface to different external analog I/O platforms supplied by Intelligent Instrumentation. It is a full-length board with 8-bit ISA bus XT-style connector.

Currently, off-the-shelf Hypersignal software (and low-level DSP32 source code) is configured to use the Intelligent Instrumentation ZPB1100 analog input board, and external ZPD 1007 analog I/O unit without any programming required. Other external modules that have not been mentioned will probably work if they go straight in to the ZPB32 serial port connectors and can provide 32-bit dual-channel input or 16-bit single-channel input. In such cases, we recommend either '-ZPD1001' or '-ZPD1007' (each having slightly different default DSP32C ioc settings) menu entries for 32-bit input, as this will activate 2-channels (both input and output, and the '-ZPB1100' menu entry for 16-bit input/output. Otherwise, non-supported analog I/O modules can still be easily interfaced by connecting them directly to the ZPB32 serial ports and then making minor modifications directly to the DSP32 source code (see section 4.2, AT&T WE DSP32 DSP Source Code Interface).

The most capable analog I/O unit for the ZPB32 is the ZPD 1007; this is the unit referred to below in the specifications list. This unit provides 16-bit, 2-channel input and output at sampling rates from 64 Hz up to a maximum of 200 kHz. The ZPD 1007 includes no anti-aliasing or reconstruction filters—see the discussion in section 3.0, HARDWARE CONSIDERATIONS, for possible remedies to this problem. The ZPB1100, on the other hand does have a 3-pole Butterworth anti-aliasing filter. However, the ZPB1100 has no D/A output; see the Intelligent Instrumentation catalog for a complete list of possible analog I/O modules (and a much better description than can be provided here).

Initial Hypersignal I/O base address for the ZPB32 is 300H, which is also the Intelligent Instrumentation factory default value.

When installing a ZPD 1007 external unit, there is only one decision you must make: whether to use the DIP switch-controlled internal sampling rate or enable the front panel external clock. Factory configuration is a switch setting that corresponds to about a 46 kHz rate; this can be changed by a DIP switch inside the ZPD 1007 unit (consult the Intelligent Instrumentation documentation).

When installing a ZPB1100 board, you must ensure that the I/O base address for the ZPB1100 is set to the I/O base address of the ZPB32 + 10H. This would be 310H, using factory default addressing. Connect the P4 header on the ZPB32 board to the H1 header on the ZPB1100 board, ensuring that the red stripe on the ribbon cable goes to pin 1 at either end. Input range (gain) and external sampling-rate options can be controlled from the software menus. Input range settings of 3 (±3 Volts full-scale), .3, .03, .015, and .006 are possible in the Digital Oscilloscope and Spectrum Analyzer functions; these same settings are enterable in the form of gain (1, 10, 100, 200, 500) in the Analog Conversion menu. Also, you can enter the '-EXT' attachment after the '-ZPB1100' entry to disable the built-in programmable sampling rate and enable the external clock input on the ZPB1100 backplane connector.

NOTES: ***For the ZPD 1007, even though Hypersignal functions allow entry of sampling rates other than the current fixed rate or external clock, actual programming of the sampling is not occurring.*** **You should enter a matching value in the software SAMPLING FREQUENCY menu fields, in order to maintain correct time units display.**

Although the ZPD 1007 is capable of sampling rates up to 200 kHz, it is unlikely that the ZPB32 can maintain continuous operation (gap-free data) at rates above about 80 kHz due to the fact that the DSP32 has only a 160 nsec instruction cycle time, and does not have a real-time interrupt structure. With these limitations, the code running on the ZPB32 is unlikely to operate reliably above 80 kHz.

The ZPB1100 board *does allow software-programmable-sampling rates.* You can optionally enable an external clock for the ZPB1100 by attaching '-EXT' to its menu entry(s) in the System Configuration menu. With an external clock operating, you should enter a matching value for Hypersignal menu SAMPLING FREQUENCY fields, in order to maintain correct time units display and usage.

D/A capability with the ZPB1100 board can be obtained by installing a ZPB2100 analog output board, and connecting the "H2" headers on the ZPB1100 and ZPB2100 together. If you are *using only the ZPB2100 board,* then you must still enter '-ZPB1100' after the relevant fields in the System Configuration menu, and you must also provide an external clock to the ZPB2100 backplane.

Input and output offset and trigger entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32 chip real-time software.

Hardware Ratings and Software Throughput Specifications

The analog I/o specifications below refer to the ZPD 1007 analog I/O unit.

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32, 160 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	55 kHz
1 channel Hard Disk Playback:	55 kHz
Max. 2-chn. Spectrum Analyzer Rate:	80 kHz
1024 pt FFT Accel. Disk File Rate:	10 frames/sec
Spectrum Analyzer Noise Floor:	85 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	16k x 32
Ext. Sync Serial Interface:	no (used for analog I/O)
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	88 dB
THD:	UA
Voltage Range:	±2.75V
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	no (DIP switch-selectable)
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	64 Hz (0 w/ ext. clock)
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	88 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	no (DIP switch-selectable)
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	64 kHz (0 w/ ext. clock)
External TTL-Level Trigger:	no

Onboard Memory Usage

The basic ZPB32 board configuration is 16k x 32 of total memory (either program or data), although there may be some boards with only 8k x 32 of memory. Hypersignal uses 1k x 32 of the memory for program storage. Hypersignal will detect and operate with either amount of memory, but the 8k x 32 configuration will result in degraded operation (including slower maximum continuous data disk transfer rate, smaller maximum FFT sizes, and reduced multichannel capability in the Spectrum Analyzer function); we highly recommend the 16k x 32 configuration.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the either the ZPB1100 or the ZPD 1007 in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

The stimulus & response function and snap-in real-time filtering function are not active for any DSP32 boards. Due to the difficulty in installing these functions on the DSP32 chip, which lacks a usable real-time interrupt structure, we have decided to not support these functions on DSP32 boards. Convolution acceleration is also not implemented on DSP32 boards. The DSP32C, on the other hand, is the "second-generation" DSP32, with many improvements; DSP32C-based boards have full functionality under Hypersignal and in fact boast some of the fastest performance numbers of all DSP chips currently supported. The "ZPB34" is the DSP32C Intelligent Instrumentation cousin to the ZPB32.

Additional Notes About Hypersignal Software Operation for ZPB1100-Based Setups

1) The board has no D/A capability by itself (see ZPB2100 disussion, above). The DSP ACCELERATOR field in the System Configuration menu takes over as the default D/A playback hardware in the Waveform Display/Edit and 2-D Spectrograph functions, allowing a second board to perform the D/A function if present in the system.

2) If you enter the '-EXT' attachment after the '-ZPB1100' entry(s) in the System Configuration menu, then the programmable sampling-rate is disabled and you will have to provide an external clock to the backplane connector on the ZPB1100. In this case, sampling rates entered in the high-level software will have no effect. Nonetheless, you should enter a value that matches your external clock in order to maintain proper time units display in the software. If you have interfaced analog I/O directly to the ZPB32, and your analog I/O subsystem is somehow capable of accepting a programmable sampling rate value, then use the value "sam_rate" in the DSP32 source code, as it is sent down by the high-level software (see section 4.2, AT&T WE DSP32 DSP Source Code Interface).

3) The INPUT GAIN field in the Analog Conversion affects the built-in *hardware* gain circuitry; the DSP32 software gain factor is set to 1. The OUTPUT GAIN field controls the software gain, as normal, since their is no D/A output on this board. In the Digital Oscilloscope and Spectrum Analyzer functions, the INPUT RANGE field affects the hardware gain circuitry; the software gain controls ('PgUp' and 'PgDn' keys) default to one, and continue to function normally.

Additional Notes About Hypersignal Software Operation for ZPD 1007-Based Setups

1) The ZPD 1007 is shipped configured for a fixed 46 kHz sampling rate; you can change this fixed rate via DIP switch inside the ZPD 1007; also, you can disable the switch and enable a front panel external clock. *Although the ZPD 1007 is capabable of sampling rates up to 200 kHz, it is unlikely that the DSP32 will operate continuously above 100 kHz, due to its 160 nsec instruction cycle time, and lack of a real-time interrupt structure.*

When using an external clock, you should enter a matching number in the Hypersignal software Analog Conversion, Digital Oscilloscope, and Spectrum Analyzer SAMPLING FREQUENCY fields, in order to maintain correct time units display and wave-

form storage. Entering the '-EXT' attachment after the '-ZPD1007' entry(s) in the System Configuration menu has no effect, since the unit has no programmable sampling-rate capability in the first place. If you have interfaced analog I/O directly to the ZPB32, and your analog I/O subsystem is somehow capable of accepting a programmable sampling rate value, then use the value "sam_rate" in the DSP32 source code, as it is sent down by the high-level software (see section 4.2, AT&T WE DSP32 DSP Source Code Interface).

2) The ZPD 1007 apparently has inverting amplifier stages prior to its A/D converters which cause "positive" input signals to be inverted. For a signal such as speech or music this probably makes no difference, but for a well-defined measurement signal (such as a square-wave), this could be a problem. If want to counter this behavior you will have to obtain the DSP32 source code and make the appropriate minor modifications (see section 4.2, AT&T WE DSP32 Source Code Interface).

3) The ZPD 1007 apparently reverses the order of samples in its 32-bit transfer-word from input to output. For example, if you enable "monitor mode" digital loopback in the Analog Conversion function, you will hear data from input 1 on output 2 and vice versa. We took a a magic marker and re-labeled our ZPD 1007 outputs.

3.11.4 I^3 ZPB34 (DSP32C)

Basic Descriptive and Installation Details

Use a hardware designator of 'ZPB34' in the Hypersignal software System Configuration menu for a ZPB34 board with no analog I/O interface boards connected, 'ZPB34-6064' for one or two (1 or 2) ZPB6064 analog I/O interface boards connected, 'ZPB34-ZPB1100' for a ZPB1100 and/or ZPB2100 board connected, 'ZPB34-ZPD1007' for a ZPD 1007 external unit connected, and 'ZPB34-ZPB100' for a ZPB100 "speech" board with T7520 codec connected. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The ZPB34 contains the 32-bit floating-point AT&T DSP32C DSP chip, and a serial interface to different external analog I/O platforms supplied by Intelligent Instrumentation. It is a full-length board with 16-bit ISA bus AT-style connector.

Currently, off-the-shelf Hypersignal software (and low-level DSP32C source code) is configured to use the Intelligent Instrumentation ZPB6064 analog I/O interface board(s), ZPB1100 analog input board, and external ZPD 1007 analog I/O unit without any programming required. This means that any Intelligent Instrumentation analog I/O modules connectible to the ZPB6064 interface boards, such as the ZPD 1002 (12-bit, 10MHz) external analog input unit, are also supported. Other external modules that have not been mentioned will probably work if they go straight in to the ZPB34 serial port connectors and can provide 32-bit dual-channel data or 16-bit single-channel data. In such cases, we recommend either '-ZPD1001' or '-ZPD1007' menu entries (each having slightly different default DSP32C ioc settings) for 32-bit input/output, and the '-ZPB1100' menu entry for 16-bit input/output. Otherwise, non-supported analog I/O modules can still be easily interfaced by connecting them directly to the ZPB34 serial ports and then making minor modifications directly to the DSP32C source code (see section 4.4, AT&T WE DSP32C DSP Source Code Interface).

The fastest Intelligent Instrumentation ZPB6064-compatible analog I/O unit is the ZPD 1002; this is the unit referred to below in the specifications list. This unit provides 12-bit, 2-channel input at sampling rates from DC up to a maximum of 10 MHz. The ZPD 1002 includes no anti-aliasing or reconstruction filters—see the discussion in section 3.0, HARDWARE CONSIDERATIONS, for possible remedies to this problem. The ZPB1100, on the other hand, does have a 3-pole Butterworth anti-aliasing filter. The ZPD 1007, like the ZPD 1002, has none. Also, the ZPD 1002 and ZPB1100 have no D/A output while the ZPD 1007 does; see the Intelligent Instrumentation catalog for a complete list of possible analog I/O modules (and a much better description than can be provided here).

The ZPB100 is a special-purpose "speech" board that can be connected directly to the P1 and P2 internal connectors on the ZPB34, providing 1 15-bit (10-bit linearity) input/output channel at a fixed sampling rate of 8 kHz, with built-in anti-aliasing and reconstruction filtering. Maximum input and output voltage ranges are ±1.578V. Maximum dynamic range is over 80dB.

Initial Hypersignal I/O base address for the ZPB34 is 300H, which is also the Intelligent Instrumentation factory default value.

There are several important considerations when installing the ZPB34 and ZPB6064 interface boards. First, Hypersignal expects the 6064 board to be at the I/O base address of the ZPB34 + 10H. *If a second 6064 is being used, it should also be at the ZPB34 I/O base address + 10H.* If you are using factory default addressing, this would be 310H. Second, you should install each ZPB6064 board with Switch 9 (SW9) set to on. This enables "dual-map mode", which allows Hypersignal to access both a single board or two boards simultaneously. *You should set SW9=ON regardless of whether you are using one or two 6064 boards.* Although the Intelligent Instrumentation manual is ambiguous in this area, setting SW9=ON disables address switches

S1-S2. Third, the first ZPB6064 board in use must be the "Master" (SW10=OFF). A second 6064 board, if in use, must be the "Slave" (SW10=ON). Finally, use the short, 16-pin ribbon cable provided by Intelligent Instrumentation to connect the ZPB34 to the ZPB6064 board(s). Start with P3 on the ZPB34, and daisy chain to P4 on each 6064. The red stripe on the cable should go to pin 1 on each connector.

When installing a ZPD 1007 external unit, there is only one decision you must make: whether to use the fixed sampling rate or enable the front panel external clock. Factory configuration is a fixed 46 kHz rate; this can be changed by a DIP switch inside the ZPD 1007 unit (consult the Intelligent Instrumentation documentation).

When installing a ZPB1100 board, you must ensure that the I/O base address for the ZPB1100 is set to the I/O base address of the ZPB34 + 10H. This would be 310H using factory default addressing. Connect the P3 header on the ZPB34 board to the H1 header on the ZPB1100 board, ensuring that the red stripe on the ribbon cable goes to pin 1 at either end. Input range (gain) and external sampling-rate options can be controlled from the software menus. Input range settings of 3 (±3 Volts full-scale), .3, .03, .015, and .006 are possible in the Digital Oscilloscope and Spectrum Analyzer functions; these same settings are enterable in the form of gain (1, 10, 100, 200, 500) in the Analog Conversion menu. Also, you can enter the '-EXT' attachment after the '-ZPB1100' entry to disable the built-in programmable sampling rate and enable the external clock input on the ZPB1100 backplane connector.

NOTES: The ZPB100 "speech" board operates at a fixed 8 kHz rate. Analog I/O modules connected to ZPB6064 interface boards will need an externally provided sampling rate clock. Similarly, the ZPD 1007 unit operates either at a switch-controlled fixed rate or also needs an external clock (jumperable option). *Thus, even though Hypersignal functions allow entry of sampling rates other than the current fixed rate, actual programming of the sampling rate in these two cases is not occurring.*

The ZPB1100 board, on the other hand, *does allow programmable-sampling rates.* If you optionally enable the external clock for the ZPB1100 by attaching '-EXT' to its menu entry(s) in the System Configuration menu, then again entry of sampling rate values in Hypersignal menus has no effect.

When using a fixed rate or external clock, you should still enter matching SAMPLING FREQUENCY values in the software menus in order to ensure correct time units display. If you interface analog hardware directly to the ZPB34 serial ports that is capable of some programming, then see the variables "Fs_mode" and "Fs_value" in the DSP32C source code, which represent the hardware register value and actual sampling rate in Hz, respectively, and are sent down by the high-level software based on menu entry.

D/A capability with the ZPB1100 board can be obtained by installing a ZPB2100 analog output board, and connecting the "H2" headers on the ZPB1100 and ZPB2100 together. If you are *using only the ZPB2100 board,* then you must still enter '-ZPB1100' after the relevant fields in the System Configuration menu, and you must also provide an external clock to the ZPB2100 backplane.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -2047 to 2047 for 12-bit A/D, and -32767 to 32767 for 16-bit A/D), and applied digitally, via DSP32C code.

Hardware Ratings and Software Throughput Specifications

NOTE: The analog I/O specifications below refer to the ZPD 1002, which is fastest unit possible with the ZPB34. The digital specifications apply regardless of analog I/O board or external unit.

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	10 MHz (non-continuous)
1 channel Hard Disk Playback:	N/A
Max. 2-chn. Spectrum Analyzer Rate:	10 MHz
1024 pt FFT Accel. Disk File Rate:	19 frames/sec
Spectrum Analyzer Noise Floor:	78 dB (@ 1MHz)

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	144k x 32
Ext. Sync Serial Interface:	no (used for analog I/O)
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 12-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	80 dB
THD:	UA
Voltage Range:	±3V
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	no
External Sample Clock:	yes
Max. Sampling Rate:	10 MHz
Min. Sampling Rate:	0 Hz
External TTL-Level Trigger:	yes
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	N/A
Codec channels:	N/A
Expandable:	N/A
Simultaneous S/H:	N/A
Typical Dynamic Range:	N/A
THD:	N/A
Voltage Range:	N/A
Output Impedance:	UA
Reconstruction Filters:	N/A
Sin(x)/x Filters:	N/A
Programmable Sampling Rate:	N/A
External Sample Clock:	N/A
Max. Sampling Rate:	N/A
Min. Sampling Rate:	N/A
External TTL-Level Trigger:	N/A

Onboard Memory Usage

The basic, no-frills ZPB34 board contains 16k x 32 of zero wait-state external A area memory. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog data buffers and file-based acceleration processing. Other versions of the ZPB34 contain additional zero wait-state external B area memory, with the minimum being 16k x 32, and the maximum 128k x 32. Hypersignal software firsts checks the external B area for any memory, and if found, places all analog buffers and processing areas there instead of external A. We strongly recommend 64k x 32 of external B area memory in order to enable all Hypersignal acceleration functions and allow continuous real-time functions to operate at full speed and capability. For example, 16k x 32 of data memory is not enough to support 2-channel 4096 pt. FFT Spectrum Analyzer operation, but will allow smaller FFT sizes and/or 1-channel operation. Also, 64k x 32 of external B memory will enable convolution/correlation acceleration, and greatly increase output stimulus sizes for stimulus & response operation.

Additional Notes About Hypersignal Software Operation for ZPB6064-Based Setups

There are several deviations from standard Hypersignal operation for the ZPB34 when used with ZPB6064 analog I/O interface boards.

1) Data-collection cannot be made continuous, due to the design of the ZPB6064 interface boards. This has several ramifications:

1a) Data collected in the Analog Conversion function will have a discontinuity (representing a gap in the data) at intervals of time equivalent to 24576 samples for single-channel waveforms and equivalent to 12288 samples for two-channel waveforms.

1b) If supported, signal playback from disk file will also have a discontinuity every 24576 samples. This will “sound” like a gap, or skip, between data segments. Signal playback includes the Analog Conversion function and playback from waveform displays.

1c) Digital filters applied in the instrumentation displays or Analog Conversion functions will show incorrect results at the beginning of each trace display. Trace display will be incorrect for a number of samples equal to the length of the filter's impulse response. This results from being unable to apply the filter to some number of points before those that have recently been collected from the 6064.

1d) Triggering cannot be performed as usual. See items 3) and 4) below for a discussion of triggering capability.

2) The system may have no D/A capability (true for the ZPD 1002 mentioned above). If so, the DSP ACCELERATOR field in the System Configuration menu takes over as the default D/A playback hardware in the Waveform Display/Edit and 2-D Spectrograph functions, allowing a second board to perform the D/A function if present in the system.

3) In the Analog Conversion function, the "EXT TRIGGER" field may be active, depending on the analog I/O module in use. An entry of 'INT' (internal) will cause trigger to occur exactly when you answer the "BEGIN ACQUISITION (Y/N)?" prompt; an entry of 'EXT' will cause the trigger source to be the external TTL-level trigger input on the analog I/O module front panel.

4) In the Digital Oscilloscope and Spectrum Analyzer instrumentation displays, the '+' and '-' keyboard trigger controls do not work as normal. Upon first entry into these functions trigger is "internal", occuring whenever the software is ready for more data. Tapping either the '+' or '-' keys will activate the external TTL-level trigger on the analog I/O module. When external trigger is active, a '-EXT' will appear after the current trigger trace readout at the lower right of the display.

5) In the Analog Conversion function, the TRIG DELAY field may be active, depending on the analog I/O module in use. This is true for the ZPD 1002, where a positive TRIG DELAY value (entered in seconds) will cause collected data to start this amount of time after trigger has occurred, and a negative value will cause data to be collected for this amount of time prior to trigger occurring. The physical limit on the 6064 board is 65536 samples, so be sure that your delay values do not exceed this. In the above discussion, "trigger occurred" is either the internal software trigger or the external analog I/O module TTL-level trigger. See 3) and 4) above for a details on how to select which is used.

6) If the analog I/O module in use requires an externally provided clock to operate, sampling rates entered in the high-level software will have no effect. If you provide an external clock input to the analog I/O module, you should then enter a matching number in the software. If you have interfaced analog I/O directly to the ZPB34 serial ports, and the analog I/O subsystem is capable of accepting a programmable sampling rate value, then see the values "Fs_mode" and "Fs_Value" in the DSP32C source code, which represent the hardware register setting and actual sampling rate in Hz, respectively, and are sent down by the high-level software.

Additional Notes About Hypersignal Software Operation for ZPB1100-Based Setups

1) The board has no D/A capability by itself. The DSP ACCELERATOR field in the System Configuration menu takes over as the default D/A playback hardware in the Waveform Display/Edit and 2-D Spectrograph functions, allowing a second board to perform the D/A function if present in the system.

2) If you enter the '-EXT' attachment after the '-ZPB1100' entry(s) in the System Configuration menu, then the programmable sampling-rate is disabled and you will have to provide an external clock to the backplane connector on the ZPB1100. In this case, sampling rates entered in the high-level software will have no effect. You should enter a value that matches your external clock in order to maintain proper time units display in the software. If you have interfaced analog I/O directly to the ZPB34, and your analog I/O subsystem is somehow capable of accepting a programmable sampling rate value, then see the values "Fs_mode" and "Fs_value" in the DSP32C source code, which represent the hardware register setting and actual sampling rate in Hz, respectively, and are sent down by the high-level software (see section 4.4, AT&T WE DSP32C DSP Source Code Interface).

3) The INPUT GAIN field in the Analog Conversion affects the built-in *hardware* gain circuitry; the DSP32C software gain factor is set to 1. The OUTPUT GAIN field controls the software gain, as normal, since their is no D/A output on this board. In the Digital Oscilloscope and Spectrum Analyzer functions, the INPUT RANGE field affects the hardware gain circuitry; the software gain controls ('PgUp' and 'PgDn' keys) default to one, and continue to function normally.

Additional Notes About Hypersignal Software Operation for ZPD 1007-Based Setups

1) The ZPD 1007 is shipped configured for a fixed 46 kHz sampling rate; you can change this fixed rate via DIP switch inside the ZPD 1007; also, you can disable the switch and enable a front panel external clock. In either case, you should enter a matching number in the Hypersignal software Analog Conversion, Digital Oscilloscope, and Spectrum Analyzer SAMPLING FREQUENCY fields, in order to maintain correct time units display and waveform storage. Entering the '-EXT' attachment after the '-ZPD1007' entry(s) in the System Configuration menu has no effect, since the unit has no software-programmable sampling-rate capability. If you have interfaced analog I/O directly to the ZPB34, and your analog I/O subsystem is somehow

capable of accepting a programmable sampling rate value, then see the values “Fs_mode” and “Fs_value” in the DSP32C source code, which represent the hardware register setting and actual sampling rate in Hz, respectively, and are sent down by the high-level software (see section 4.4, AT&T WE DSP32C DSP Source Code Interface).

3.12 Loughborough Sound Images (LSI) / Spectrum Signal Processing

Loughborough Sound Images (LSI; United Kingdom) boards, distributed and enhanced by Spectrum Signal Processing in North America, are well-known as standardized platforms for algorithm development and real-time signal processing development across several DSP chip families. These boards are usually quite flexible in terms of DSP development, memory organization and expansion, analog I/O capability and expansion, and DSP serial port accessibility. They provide very fast and straightforward host interfaces, and good support development utilities (debuggers, etc).

The “DSPLink” parallel-interface connector on all LSI boards can be used to interface to other “add-in” cards containing additional analog channels. Modification of the DSP Source Code Interface for the appropriate DSP chip (see section 4.0, DSP SOURCE CODE INTERFACE) allows use of these additional analog channels in the standard high-level Hypersignal host software.

Multiple LSI system boards can be used in the same system, as long as each is dedicated to separate real-time or acceleration functions. In this case, Hypersignal expects the system boards to reside at the entered I/O BASE ADDRESS plus multiples of 10H. For example, if board 1 is at 290H, then board 2 would be at 2A0H (and board 3 at 2B0H, etc). Note that the system has no way of knowing if two boards from the same manufacturer are actually present in the system unless they are both selected at least once in the ANALOG CONVERSION, DSP ACCELERATOR, DIGITAL SCOPE, and SPECTRUM ANALYZER hardware functions. If you have jumpered two boards at base address and base address + 10H, and then mention only the second board in the above functions, the program will incorrectly attempt to locate this board at the base address. In this discussion, “board 1" is the first board to appear on the menu (from the top down), ”board 2" the next, etc.

3.12.1 LSI/Spectrum TMS320C25 System Board

Basic Descriptive and Installation Details

Use a hardware designator of ‘LSI320’ for the 40 MHz board and ‘LSI320-B’ for the 50 MHz board in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to ‘Y’) if you have a 80286, 80386, or 80486 machine.

The LSI TMS320C25 board is shipped standard with one 16-bit A/D and one 16-bit D/A channel. It is a full-length board with 8-bit ISA bus XT-style connector.

The maximum sampling rate for the TMS320C25 board is programmable up to 58 kHz for A/D input and up to 100 kHz for D/A output (maximum disk transfer rate will depend on amount of onboard memory; see Memory Usage paragraph below). The TMS320C25 board has *no anti-aliasing filters;* some suggestions to remedy this problem are discussed in section 2.0, HARDWARE CONSIDERATIONS, above.

Initial default Hypersignal I/O base address for all LSI boards is 290H, which is also the LSI initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

External clocking of individual sample conversions is available, but no external TTL or hardware-based variable-level trigger exists. However, customers have added this capability in the past with hardware modifications—contact Hyperception or Signalogic for suggestions.

An entry of ‘-EXT’ after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C25 code.

Jumper (Link) List for LSI TMS320C25 Board

LK1 – LK4	set for memory size
LK5	OUT
LK6	IN
LK7	OUT
LK8	IN
LK9	OUT

NOTE: LK10 & LK11 must be changed from factory configuration.

LK10	OUT (no pins connected)
LK11	OUT (no pins connected)
LK12 (a)	far left IN [| : : : : : : :]
LK13	BOTH IN [| |]
LK14	IN

NOTE: LK15 - LK17 must be changed from factory configuration (extra plugs can be found on LK10, LK11, and LK20).

LK15 (a)	[——] [· ·] [· ·]
LK16 (a)	[—- ·]
LK17 (a)	[—- ·]
LK18 (a)	[—- ·]
LK19 (b)	[· —-]
LK20	OUT
LK21	IN top two [|] [:]

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C25, 100 nsec
Data Format, Wordlength:	fixed-point, 16-bit
1 channel Hard Disk Record:	58 kHz
1 channel Hard Disk Playback:	58 kHz
Max. 2-chn. Spectrum Analyzer Rate:	58 kHz
1024 pt FFT Accel. Disk File Rate:	10.6 frames/sec
Spectrum Analyzer Noise Floor:	67 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 16 P; 8k x 16 D
Maximum Ext. Mem. Config:	32k x 16 P; 32k x 16 D
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPLink
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	1, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H	no
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±10V
Input Impedance:	UA
Anti-Aliasing Filters:	no
Programmable Sampling Rate:	no
External Sample Clock:	yes
Max. Sampling Rate:	58 kHz
Min. Sampling Rate:	76.3 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H:	no
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±10V
Output Impedance:	UA
Reconstruction Filters:	no
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	100 kHz
Min. Sampling Rate:	76.3 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The LSI TMS320C25 board is shipped standard with 8k x 16 of program memory and 8k x 16 of data memory. A memory upgrade to 32k x 16 of *both program and data* memory improves continuous transfer rates to/from disk, enables convolution/correlation acceleration, enables stimulus & response operation, and increases maximum 2-channel Spectrum Analyzer FFT size to 2048 pt.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the LSI TMS320C25 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.12.2 LSI/Spectrum DSP56001 System Board

Basic Descriptive and Installation Details

Use a hardware designator of 'LSI56' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The LSI DSP56001 board is shipped standard with two 16-bit A/D channels and two 16-bit D/A channels. It is a full-length board with 8-bit ISA bus XT-style connector.

Two-channel input and two-channel output is available at sampling rates up to 200 kHz on each channel (but continuous transfer rates to/from disk will be less—see specifications below). Two-channel operation is supported by all Hypersignal analog functions including Analog Conversion, Digital Oscilloscope, Spectrum Analyzer, and D/A output from time domain and frequency domain displays. Two-channel triggering combinations (e.g. using one channel solely for triggering, and the other for data) are supported. Input and output channels have resistor-programmable, 3rd-order Chebyshev anti-aliasing and reconstruction filters. In addition, the board contains an 8-bit 8 kHz codec, which is connected to the DSP56001 SSI serial port.

Initial default Hypersignal I/O base address for all LSI boards is 290H, which is also the LSI initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

External clocking of individual sample conversions is available, but no external TTL or hardware-based variable-level trigger exists. However, customers have added this capability in the past with hardware modifications—contact Hyperception or Signalogic for suggestions.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP56001 code.

Jumper (Link) List for LSI DSP56001 Board

LK1	[· —-] external clock disabled
LK2	[· —-] onboard timer enabled
LK3	[—- ·] [—- ·] [—- ·] b a c Y: mem X: mem P: mem
LK4	NONE (no PC interrupts used)
LK5	[—- ·] codec μ-law selected
LK6	a [|] sample/hold enabled [|] b [·]
LK7	b [·] [|] codec *output* disabled a [|] from channel 2

LK8	[—- ·] codec *input* disabled from channel 2
LK9	c [. .] sample clock provides b [——] IRQB- interrupt to a [. .] DSP56001
LK10	IN
LK11	IN
LK12*	[: | | : | : | |]

*Factory default I/O base address is 290H (note that the LK12 jumper must be read right-to-left, is in one's complement form, and actually specifies base *paragraph* address).

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP56001, 100 nsec
Data Format, Wordlength:	fixed-point, 24-bit
1 channel Hard Disk Record:	110 kHz
1 channel Hard Disk Playback:	110 kHz
Max. 2-chn. Spectrum Analyzer Rate:	76 kHz
1024 pt FFT Accel. Disk File Rate:	14.8 frames/sec
Spectrum Analyzer Noise Floor:	87 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 24 P, X, Y
Maximum Ext. Mem. Config:	64k x 24 P, X, Y
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes, DSPLink
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	1, 8kHz, 13-bit, Intel 29C13
Expandable:	yes, DSPLink
Simultaneous S/H	yes
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±3V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby3; RP
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152.6 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	1, 8kHz, 13-bit Intel 29C13
Expandable:	yes, DSPLink
Simultaneous S/H:	yes
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby3; RP
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152.6 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The LSI DSP56001 board is shipped standard with 16k x 24 of program, X: data, and Y: data memory. Hypersignal uses less than 4k x 24 of program memory, and the full 32k x 24 of data memory for continuous buffers and file-based acceleration processing areas. An upgrade to 64k x 24 of X: and Y: data memory would greatly increase stimuli size for stimulus & response operations, and enable convolution/correlation acceleration.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the LSI DSP56001 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

This board has a strange hardware design quirk that prevents single-channel output on channel 1. Single-channel input/output on channel 0, single-channel input on either channel, and dual-channel input/output all work correctly.

3.12.3 LSI/Spectrum DSP32C System Board

Basic Descriptive and Installation Details

Use a hardware designator of 'LSI32C' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The LSI DSP32C system board has two 16-bit input and output channels with up to 200 kHz sampling rates. It is a full-length board with 16-bit ISA bus AT-style connector.

However, maximum continuous transfer rate to/from hard disk for single channel operation could be as high as 150 kHz (for 33 MHz 80386 machine with 28 msec average-access-time hard drive). Two-channel operation is supported by all Hypersignal analog functions including Analog Conversion, Digital Oscilloscope, Spectrum Analyzer, and D/A output from time domain and frequency domain displays. Two-channel triggering combinations (e.g. using one channel solely for triggering, and the other for data) are also supported. Input and output channels have resistor-programmable, 4th-order Chebyshev anti-aliasing and reconstruction filters.

Initial default Hypersignal I/O base address for all LSI boards is 290H, which is also the LSI initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

External clocking of individual sample conversions is available, but no external TTL or hardware-based variable-level trigger exists. However, customers have added this capability in the past with hardware modifications—contact Hyperception or Signalogic for suggestions.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32C code.

NOTE: The current, optimized version of Hypersignal DSP32C code will not run reliably on LSI DSP32C boards with "R32" revision DSP32C chips. *You must have an "R33" revision chip.* Functions that require analog I/O are especially sensitive to memory-timing and host port operation problems related to the "R32" revision chip.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	110 kHz
1 channel Hard Disk Playback:	110 kHz
Max. 2-chn. Spectrum Analyzer Rate:	190 kHz
1024 pt FFT Accel. Disk File Rate:	20.7 frames/sec
Spectrum Analyzer Noise Floor:	82 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 32; 32k x 32 2 wait-state
Maximum Ext. Mem. Config:	8k x 32; 512k x 32 2 wait-state
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPLink
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H	yes
Typical Dynamic Range:	85 dB
THD:	.05%
Voltage Range:	±2.5V to ±3.5V; J3
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby4; RP
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152.6 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H:	yes
Typical Dynamic Range:	85 dB
THD:	.05%
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby4;RP
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152.6 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The DSP32C system board is shipped standard with 8k x 32 of program memory (addressed in external B area, 0 wait-states) and 32k x 32 of data memory (addressed in external A area, SRDYN-controlled wait-states). Hypersignal uses less than 4k x 32 of the external A area for program storage. The large amount of data memory means that all board-related functions in Hypersignal are active and operate with maximum speed and size, including the Spectrum Analyzer, which can perform 2-channel 4096 pt. FFTs. The only significant benefit to a memory upgrade for external B would be larger stimulus sizes for the Stimulus & Response function.

NOTE: Hypersignal DSP32C code sets the external B memory for 1 wait-state, even though it is supposed to be controlled by the external SRDYN signal. This was done to enable 200 kHz operation in the instrumentation displays; no adverse effects have been detected so far. Restoring SRDYN control over the wait-states would required only a minor modification to the DSP32C source code (see section 4.4, AT&T WE DSP32C DSP Source Code Interface).

Currently, external triggering operation is inactive with the LSI DSP32C board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.12.4 LSI/Spectrum TMS320C30 System Board

Basic Descriptive and Installation Details

Use a hardware designator of 'LSIC30' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The LSI TMS320C30 board is shipped standard with two 16-bit A/D channels and two 16-bit D/A channels. It is a full-length board with 16-bit ISA bus AT-style connector.

Two-channel input and two-channel output is available at sampling rates up to 200 kHz on each channel (but continuous disk transfer rates will be less—see specifications below). Two-channel operation is supported by all Hypersignal analog functions including Analog Conversion, Digital Oscilloscope, Spectrum Analyzer, and D/A output from time domain and frequency domain displays. Two-channel triggering combinations (e.g. using one channel solely for triggering, and the other for data) are also supported. Input and output channels have resistor-programmable, 4th-order Chebyshev anti-aliasing and reconstruction filters.

Initial default Hypersignal I/O base address for all LSI boards is 290H, which is also the LSI initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

External clocking of individual sample conversions is available, but no external TTL or hardware-based variable-level trigger exists. However, customers have added this capability in the past with hardware modifications—contact Hyperception or Signalogic for suggestions.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C3x code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C30, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	85 kHz
1 channel Hard Disk Playback:	85 kHz
Max. 2-chn. Spectrum Analyzer Rate:	118 kHz
1024 pt FFT Accel. Disk File Rate:	17.1 frames/sec
Spectrum Analyzer Noise Floor:	85 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	64k x 32; 64k x 32 data-only
Maximum Ext. Mem. Config:	128k x 32; 128k x 32 data-only
Ext. Sync Serial Interface:	yes (2 ports)
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPLink, memory interface
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H	yes
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±3V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby4;RP
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152.6 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H:	yes
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby4; RP
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	152.6 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The LSI TMS320C30 system board is shipped standard with 64k x 32 of main memory (both program and data) and 64k x 32 of dual-port memory (also both program and data). This is as much as any DSP board currently supported by Hypersignal. All real-time and acceleration functions are enabled and operate at maximum rates and capacity with this much memory.

NOTE: Spectrum Signal Processing university program boards may be shipped with only 16k x 32 of dual-port memory. This amount limits two-channel Spectrum Analyzer FFT sizes, stimulus size in stimulus & response operations, continuous disk transfer rates, and also disables convolution/correlation acceleration. An upgrade to 64k x 32 of dual-port memory is highly recommended.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the LSI TMS320C30 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.12.5 LSI/Spectrum ADSP-2101 System Board

Basic Descriptive and Installation Details

Use a hardware designator of 'LSI2101' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The ADSP-2101 system board contains the 16-bit fixed-point ADSP-2101 DSP, and has two analog I/O channels. It is a 2/3 length board with 16-bit ISA bus AT-style connector.

The analog circuitry of the basic LSI ADSP-2101 board relies on two Texas Instruments TLC32044 AICs (analog interface chip) connected to the two ADSP-2101 "SPORT" serial channels to provide two channels of 14-bit A/D and D/A. The TLC32044 device allows the sampling rate to be programmed, 3 levels of input gain to be selected, and contains anti-aliasing and reconstruction bandpass filters that track with the programmed sampling rate. The sampling rate and gain are controllable from Hypersignal menus. One should be aware that programming of the sampling rate is not exact, because the TLC3204x de-

vice uses two 6-bit integer counters as "divisors" into a master clock frequency, which allows only a discrete number of possibilities. Normally, input levels can be selected to be ±1.5V, ±3V, and ±6V. Also, the highpass portion of the bandpass filters can be selectively disabled (although this must be done through the Hypersignal configuration file as opposed to data-entry menus). For TLC32044 devices, a sin(x)/x D/A output filter is also incorporated.

Initial default Hypersignal I/O base address for all LSI boards is 290H, which is also the LSI initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

External clocking of individual sample conversions is available, but no external TTL or hardware-based variable-level trigger exists. However, customers have added this capability in the past with hardware modifications—contact Hyperception or Signalogic for suggestions.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

One unusual feature of the ADSP-2101 system board is that it can accept an EPROM device. The DSP can be booted from EPROM and application-specific data can be stored.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -8191 to 8191 for 14-bit A/D) and applied digitally, via ADSP-2101 code.

Jumper (Link) List for LSI ADSP-2101 Board

LK1	[: | | : | : | |]
LK2	[—- ·] PC plug-in mode
LK3	[—- ·] 8k x 8 RAM device
LK4	[· —-] 8k x 8 RAM device
LK5	[. —-] 8k x 8 RAM device
LK6	[—- ·] Flag In disabled
LK7	[· —-] Flag In from Ctrl Reg
LK8	[—- ·] SPORT 0 to TLC32044 0
LK9	NONE (SPORT 0 all outputs)
LK10	NONE (SPORT 1 all outputs)
LK11	[: : |] SPORT 1 to TLC32044 1
LK12	NONE (no PC interrupts)
LK13	[· —-] reset from Ctrl Reg

LK14 [—- ·] PC Bus Request enabled

LK15 [· —-] SPORT 1 to TLC32044 1

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	ADSP-2101, 80 nsec
Data Format, Wordlength:	fixed-point, 16-bit
1 channel Hard Disk Record:	19.2 kHz
1 channel Hard Disk Playback:	19.2 kHz
Max. 2-chn. Spectrum Analyzer Rate:	19.2 kHz
1024 pt FFT Accel. Disk File Rate:	UA
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	4k x 24 P; 4k x 16 D
Maximum Ext. Mem. Config:	16k x 24 P; 12k x 16 D
Ext. Sync Serial Interface:	yes (two, sel. to AICs or user)
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPLink
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 14-bit TLC3204x AIC
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H	yes
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±1.5V to ±6V; P3 (AIC)
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby7, JB (AIC)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 14-bit TLC3204x AIC
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H:	yes
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±1.5V to ±6V; P3 (AIC)
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby7, JB (AIC)
Sin(x)/x Filters:	yes, AIC (TLC32044)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The LSI ADSP-2101 system board is shipped standard with 4k x 24 of program memory and 4k x 16 of data memory. Hypersignal uses virtually all of the program memory for program storage, and all of the data memory for normal DSP processing and operation. The memory expansion option for this board is highly recommended. The expansion option increases program memory to 16k x 24, allowing for user-defined algorithm development, and increases data memory to 12k x 16 (but not to 16k x 16, because the ADSP-2101 reserves the upper 4k x 16 of its total address space for onchip memory and peripherals). The

12k x 16 data memory amount increases the maximum speed of Hypersignal functions such as continuous acquisition and stimulus & response (and allows for bigger stimuli sizes in this function), enables acceleration of the Convolution/Correlation function, and also increases the maximum FFT size and/or number of channels possible in the Spectrum Analyzer function and the maximum timebase in the Digital Oscilloscope function.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the LSI ADSP-2101 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored). Acceleration of the Convolution function cannot be achieved with this board because at least 16K of DSP data memory is required, and a maximum of only 12K is possible on the LSI ADSP-2101 board.

3.12.6 LSI/Spectrum ADSP-2100 System Board

Basic Descriptive and Installation Details

Use a hardware designator of 'LSI2100' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The ADSP-2100 system board contains the 16-bit fixed-point ADSP-2100 DSP, and has one analog I/O channel. It is a full-length board with 16-bit ISA bus AT-style connector.

Initial default Hypersignal I/O base address for all LSI boards is 290H, which is also the LSI initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

External clocking of individual sample conversions is available, but no external TTL or hardware-based variable-level trigger exists. However, customers have added this capability in the past with hardware modifications—contact Hyperception or Signalogic for suggestions.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via ADSP-210x code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	ADSP-2100, 125 nsec
Data Format, Wordlength:	fixed-point, 16-bit
1 channel Hard Disk Record:	UA
1 channel Hard Disk Playback:	UA
Max. 1-chn. Spectrum Analyzer Rate:	UA
1024 pt FFT Accel. Disk File Rate:	UA
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 24 P; 8k x 16 D
Maximum Ext. Mem. Config:	32k x 24 P; 32k x 16 D
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPLink
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	1, 16-bit
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H	no
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	UA
Input Impedance:	UA
Anti-Aliasing Filters:	UA
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	58 kHz
Min. Sampling Rate:	76.3 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 16-bit
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H:	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	UA
Output Impedance:	UA
Reconstruction Filters:	UA
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	58 kHz
Min. Sampling Rate:	76.3 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The LSI ADSP-2100 system board is shipped standard with 8k x 24 of program memory and 8k x 16 of data memory. Hypersignal uses about 4k x 24 of the program memory for program storage, and all of the data memory for normal DSP processing and operation. The memory expansion option for this board is highly recommended. The expansion option increases data memory to 32k x 16. This amount increases the maximum speed of Hypersignal functions such as continuous acquisition and stimulus & response (and allows for bigger stimuli sizes in this function), and also increases the maximum FFT size and/or number of channels possible in the Spectrum Analyzer function and the maximum timebase in the Digital Oscilloscope function.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the LSI ADSP-2100 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.12.7 LSI/Spectrum DSP96002 System Board

Basic Descriptive and Installation Details

Use a hardware designator of 'LSI96' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The LSI DSP96002 system board contains the 32-bit floating-point Motorola DSP96002 DSP chip, along with two 16-bit analog I/O channels. It is a full-length board with 16-bit ISA bus AT-style connector.

Analog input uses sigma-delta A/D converters, which provide programmable sampling rates, built-in anti-aliasing filters that track exactly with the programmed sampling rates, and linear phase. Analog output includes programmable sampling rates and resistor-programmable 4th-order Butterworth reconstruction filters. The programmable sampling rates on this board must be one of 16 discrete values in a range from 8 to 100 kHz allowed by combinations of four onboard crystal oscillators. Values such as 8, 50, and 100 kHz, as well as popular audio rates such as 44.1 kHz, are possible. In addition, Hypersignal supports 12-bit 128, 160, 176.4, 200, 256, 329, 352.8, and 400 kHz input sample modes (although data continuity in any disk-based operations is unlikely at these rates). Please note that since the output does not use the sigma-delta technology, the 12-bit mode and rates above 100 kHz do not apply to output operation.

Initial default Hypersignal I/O base address for all LSI boards is 290H, which is also the LSI initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address. Initial default Hypersignal memory base address for the DSP96002 system board is DC00H. If you need to change this value from DC00H, then you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") file directly. See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the LSI DSP96002 board memory base address in the "HSMACRO.CNF" (or "HSACOUS.CNF") file. Note also that the memory base address is never actually set via jumpers (links) or switches on the board (as is the I/O base address), but is instead written to registers on the board via high-level Hypersignal software.

NOTE: We highly suggest you not change the default value of DC00H for memory base address. This value will allow the board to work with network cards and/or "SuperVGA" video cards requiring extra "high memory". This value also preserves E000H-EFFFH as a possible EMS memory segment.

Also, be sure that you "block out" the region from DC00-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the DSP96002 system board.

MORE NOTES: The first revision of the LSI DSP96002 system board has an analog glitch caused by a problem in PCB layout. The glitch is data-dependent (basically being caused by databus energy being transferred inadvertently to an important analog "framesync" signal), and manifests as "spikes" on analog data. This glitch can be fixed by connecting 330 Ohm pull-down and 220 Ohm pull-up resistors (1 each) to pins 11 and 13 of the backplane DB15 connector. We suggest using feedthroughs located near the ground and +5V terminals on the PC I/O bus edge connector of the board.

Ensure that IRQC is jumpered as the host PC interrupt when using Hypersignal. IRQC is used by the host PC to initiate complex onchip DMA transfers involving the onboard dual-port memory (in external X: memory). More details can be found in section 4.8, Motorola DSP96002 DSP Source Code Interface.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP96002 code.

Jumper (Link) List for LSI DSP96002 Board

LK1	[| : :] /IRQA from DSPLink
LK2	[| : :] /IRQB from analog
LK3	[| : :] /IRQC from host PC
LK4	[—-] Sync. analog clock enabled
LK5	[!] Port A 32k x 32
LK6	[!] Port A in non-overlay mode
LK7	[: | :] Port A one wait-state

LK8	[—-] IN
LK9	[—- ·] Channel A single-ended
LK10	[· —-] Channel A single-ended
LK11	[· —-] Channel A single-ended
LK12	[—- ·] Channel B single-ended
LK13	[· —-] Channel B single-ended
LK14	[—- ·] Channel B single-ended
LK15	[: | | : | : | |] I/O Base Address*

*Factory default I/O base address is 290H (note that the LK15 jumper must be read right-to-left, is in one's complement form, and actually specifies base *paragraph* address).

LK16	[: : : : : :] No interrupt to host PC
LK17	[—-] Sync. analog frame enabled
LK18	[· —-] Port B 32k x 32
LK19	[|] Dual-port RAM in C00000H-DFFFFFH range
LK20	[: : : :] No Motoway interrupt
LK21	[: : : |] Bus grant level 3
LK22	[: : : |] Bus request level 3
LK23	[—- ·] Motoway arbiter enabled
LK24	[—- ·] Motoway board address 0
LK25	[: | :] Port B one wait-state
LK26	[: : |] Motoway one wait-state
LK27	[· —-] Port B non-overlay mode
LK28	[—- ·] Arbiter set to round-robin
LK29	[: : : :] No Motoway int. source

LK7 and LK25 are shown as indicating 1 wait-state above in case 128k x 32 memory devices are currently or eventually installed. If you are sure that only 32k x 32 devices will be used, you can safely set these jumpers for zero wait-states.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP96002, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	100 kHz
1 channel Hard Disk Playback:	100 kHz
Max. 2-chn. Spectrum Analyzer Rate:	128 kHz
1024 pt FFT Accel. Disk File Rate:	26 frames/sec
Spectrum Analyzer Noise Floor:	UA dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	160k x 32 A, 160k x 32 B
Maximum Ext. Mem. Config:	544k x 32 A, 544k x 32 B
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, DSPLink
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±1V SI, or ±2V DI
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P24
External Sample Clock:	yes
Max. Sampling Rate:	100 kHz 16-bit, 400 kHz 12-bit
Min. Sampling Rate:	8 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	yes, DSPLink
Simultaneous S/H:	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±1V
Output Impedance:	UA
Reconstruction Filters:	yes, Butter4; RP (default Fc=18.5 kHz)
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes, P16
External Sample Clock:	yes
Max. Sampling Rate:	100 kHz
Min. Sampling Rate:	8 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The LSI DSP96002 system board is shipped standard with 160k x 32 of port A static RAM (SRAM) memory and 160k x 32 of port B SRAM. The use of "port" terminology with the DSP96002 chip means two physically separate sets of instruction and data buses. Furthermore, Hypersignal always uses a "vector" memory configuration within each port, since this provides for fastest execution of most DSP programs. In this configuration, Hypersignal configures port A as 40k x 32 of external X: memory and 40k x 32 of external Y: memory, with both starting at 200H (just after internal data memory). Port B is then configured as 80k x 32 of external program memory, starting at 400H (just after internal program memory). This configuration offers several advantages:

1) depending on the speed of the devices, all external memory is SRAM, which can be made zero wait-state

2) external data memory is set up in true vector fashion, with X: and Y: areas being independently accessible, and starting at the same base address

3) external data memory and external program memory are on separate physical buses, allowing the DSP96002 to fetch the next instruction at the same time as data, improving throughput and processing speed

Hypersignal uses less than 8k x 32 of the program memory (which can be expanded to 272k x 32). Because the external data memory amounts are relatively large, all Hypersignal analog I/O and acceleration function operate to their maximum capability in the standard DSP-96 configuration. The only function that could benefit from data memory expansion (which could be up to a maximum of 136k x 32 of X: and Y: each) is the stimulus & response operation, which could output larger stimuli waveforms.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the LSI DSP96002 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.13 MTT Instrumentation

MTT Instrumentation offers the "DSP4xxx" series of DSP and analog I/O systems for the NEC PC-980x and compatible series of machines used frequently in Japan and elsewhere throughout Asia. The DSP4xxx series of systems are comprised of a "DSP4x00" DSP board (DSP, memory, external serial port(s), interface circuitry, etc.), a "DSP4x10" multiple-channel analog input board, an optional "DSP4x20" analog output multiple-channel board, an optional "DSP4x40" programmable clock board responsible for generating analog I/O sampling frequencies, and an optional "DSP4x50" filter board, containing anti-aliasing filters to be used with the other analog I/O boards. If the clock board is not used, then an "interior oscillator" located on the DSP4x10 analog input board can be enabled instead. Also available is a "DSP4x30" digital I/O board that can be used to enable the DSP device to access external devices and instruments for control and status purposes, and also provide external TTL level trigger and clock signals for analog I/O operation.

Currently, the DSP4100 system, based on the AT&T DSP32C floating-point device, and the DSP4200 system, based on the Texas Instruments TMS320C30 floating-point device, are supported by Nihon version Hypersignal-Macro, Hypersignal-Macro EX, and Hypersignal-Acoustic software.

3.13.1 DSP4100 System

Basic Descriptive and Installation Details

Use a hardware designator of 'MTT32C' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine (except as noted below).

The MTT Instrumentation DSP4100 system is based on the AT&T DSP32C 32-bit floating-point device. When used with a DSP4010 or DSP4110 analog input board, the system can offer eight input channels with a maximum sampling rate up to 200 kHz, and minimum sampling rates down to 1/1200 Hz. When used with a DSP4x20 analog output board, the system can additionally offer eight output channels with a maximum sampling rate up to 200 kHz, and minimum sampling rates down to 1/1200 Hz. The DSP4100 is a full-length board compatible with PC-98xx series machines.

With the DSP4010 and DSP4110 analog I/O boards, all sampling is synchronous, with simultaneous sample-and-hold. When using Hypersignal, the maximum continuous transfer rate to/from hard disk for single channel operation should be about 110 kHz (for 25 MHz 80386 machine with 28 msec average-access-time hard drive). This means that the maximum eight-channel sampling rate would be about 15 kHz per channel. Currently, two-channel operation (analog channels 1 and 2) is supported by all Hypersignal analog functions with no programming required, including Analog Conversion, Digital Oscilloscope, Spectrum Analyzer, and D/A output from time domain and frequency domain displays. Two-channel triggering combinations (e.g. using one channel solely for triggering, and the other for data) are also supported. If the DSP4x40 clock board has not been installed, then Hypersignal will default to use of the "interior oscillator" located on the DSP4x10 analog input board. In either case, note that the desired (sampling rate programmed into Hypersignal menus) and the actual sampling rate may not be the same, due to the finite possibilities available; however, Hypersignal chooses the closest available rate. Also, note that the DSP4x20 analog output board contains no onboard oscillator, and thus requires the presence of either the DSP4x40 clock board or an analog input board to operate.

3.13.1 DSP4100 System

Initial default Hypersignal I/O base address for all MTT Instrumentation DSP4x00 systems is D2H, which is also the MTT Instrumentation initial factory default. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

NOTE: For Nihon versions of Hypersignal software, the System Config menu shows has only an MTT subfield in the I/O BASE ADDRESS field. No other manufacturers are currently listed. The examples in Users Manual section 2.5.2 may show several different manufacturers, which should be ignored.

External clocking of individual sample conversions and external TTL-level trigger can be implemented using external inputs on the DSP4x40 clock board or the DSP4x30 digital I/O board. Software-initiated clocking of individual samples is also available. No hardware-based variable-level analog trigger exists.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32C code.

NOTES: 1) The current, optimized version of Hypersignal DSP32C code may not run reliably on MTT Instrumentation DSP4100 boards with "R32" revision DSP32C chips. *You should have an "R33" revision chip*. Functions that require analog I/O are especially sensitive to memory-timing and host port operation problems related to the "R32" revision chip. You may need to consult with MTT Instrumentation about the chip revision number, since the DSP4100 DSP32C device may be physically obscured by a heat sink.

2) Although the instructions above indicate that the 80286 INSTRUCTION SET field should be enabled (set to 'Y') when using the DSP4100 system, it may be necessary to disable this field (set to 'N') during D/A output operation. Evidently, the type of fast string output instructions used on 80286 and later host computer microprocessors can cause digital noise to appear on the analog outputs of the DSP4x20 board, if host port communications between the host and DSP32C chip are extensive at the same time. Functions affected include D/A output from waveform displays, and continuous signal generation (playback) from disk. Note that stimulus & response operation is not affected, because the output stimulus is pre-loaded, and thus no fast string output instructions are executed by the host during actual measurement operation. Disabling the 80286 INSTRUCTION SET field will eliminate the analog output digital noise. Input operations are not affected.

3) In its default configuration, the Hypersignal DSP32C source code expects the INTREQ1- signal to cause analog I/O interrupts for the MTT DSP4100 system. This can be changed by modifying the source code directly; see section see section 4.4, AT&T WE DSP32C DSP Source Code Interface.

Hardware Ratings and Software Throughput Specifications

The specifications below assume a system consisting of a DSP4x10 analog input board, DSP4x20 analog output board, and a DSP4x40 clock board.

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	110 kHz
1 channel Hard Disk Playback:	110 kHz
Max. 2-chn. Spectrum Analyzer Rate:	190 kHz
1024 pt FFT Accel. Disk File Rate:	21 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	128k x 32 0 wait-state
Maximum Ext. Mem. Config:	UA
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	no
External Parallel Interface:	yes, external connector
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	8, 12-bits
Codec channels:	none
Expandable:	UA
Simultaneous S/H	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±10V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, if DSP4x50 filter board used
Programmable Sampling Rate:	yes
External Sample Clock:	yes, via DSP4x30 digital I/O board or DSP4x20 clock board
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	1/1200 Hz
External TTL-Level Trigger:	yes, via DSP4x30 digital I/O board
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	8, 14-bits
Codec channels:	none
Expandable:	UA
Simultaneous S/H:	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±10V
Output Impedance:	UA
Reconstruction Filters:	yes, if DSP4x50 clock board used
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes, via DSP4x30 digital I/O board or DSP4x20 clock board
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	1/1200 Hz
External TTL-Level Trigger:	yes, via DSP4x30 digital I/O board

Onboard Memory Usage

The DSP4100 system board is shipped standard with 128k x 32 of combined program and data memory (addressed in external A area, 0 wait-states). Hypersignal uses less than 4k x 32 of the external A area for program storage. The remaining amount is used for processing and analog I/O data buffers. The large amount of data memory means that all board-related functions in Hypersignal are active and operate with maximum speed and size, including the Spectrum Analyzer, which can perform 2-channel 4096 pt. FFTs. The only significant benefit to a memory upgrade for the external A area would be larger output stimulus sizes for the Stimulus & Response measurement function.

NOTE: Hypersignal DSP32C code sets the external B memory for 2 or more wait-states, controlled by the external SRDYN signal. This is done in order to accommodate DSP4x00 system external analog and digital I/O boards, which are memory-mapped into this area.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the MTT Instrumentation DSP4100 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.13.2 DSP4200 System

Basic Descriptive and Installation Details

Use a hardware designator of 'MTTC30' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The MTT Instrumentation DSP4200 system is based on the Texas Instruments TMS320C30 32-bit floating-point device. When used with a DSP4010 or DSP4110 analog input board, the system can offer eight input channels with a maximum sampling rate up to 200 kHz, and minimum sampling rates down to 1/1200 Hz. When used with a DSP4x20 analog output board, the system can additionally offer eight output channels with a maximum sampling rate up to 200 kHz, and minimum sampling rates down to 1/1200 Hz. The DSP4200 is a full-length board compatible with PC-98xx series machines.

Hypersignal software makes use of FIFOs located on the DSP4200 board to allow real-time, continuous transfer of data regardless of whether the TMS320C30 is running or not. A small, "host-interface" program is downloaded to the DSP4200 board after reset, while the processor is still held in reset. Then, from this point forward, a small, fast interrupt service routine is used for all subsequent program and data transfers, using the onboard FIFOs in all cases. This technique enables Hypersignal to access all resources on the DSP4200, including external SRAM, onchip RAM, and I/O peripherals.

With the DSP4010 and DSP4110 analog I/O boards, all sampling is synchronous, with simultaneous sample-and-hold. When using Hypersignal, the maximum continuous transfer rate to/from hard disk for single channel operation should be about 100 kHz (for 25 MHz 80386 machine with 28 msec average-access-time hard drive). This means that the maximum eight-channel sampling rate would be about 12.5 kHz per channel. Currently, two-channel operation (analog channels 1 and 2) is supported by all Hypersignal analog functions with no programming required, including Analog Conversion, Digital Oscilloscope, Spectrum Analyzer, and D/A output from time domain and frequency domain displays. Two-channel triggering combinations (e.g. using one channel solely for triggering, and the other for data) are also supported. If the DSP4x40 clock board has not been installed, then Hypersignal will default to use of the "interior oscillator" located on the DSP4x10 analog input board. In either case, note that the desired (sampling rate programmed into Hypersignal menus) and the actual sampling rate may not be the same, due to the finite possibilities available; however, Hypersignal chooses the closest available rate. Also, note that the DSP4x20 analog output board contains no onboard oscillator, and thus requires the clock board to operate.

Initial default Hypersignal I/O base address for all MTT Instrumentation DSP4x00 systems is D2H, which is also the MTT Instrumentation initial factory setting. The DSP4200 system also use a memory base address for MTT software utilties, but Hypersignal does not currently make use of a host PC memory base address. For future reference only, a DSP4200 memory base address can be set in the System Config menu. It has a default value of D000H for both Hypersignal and the MTT Instrumentation initial DSP4200 factory setting. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS and MEM BASE ADDRESS fields, for details on changing this address.

NOTE: For Nihon versions of Hypersignal software, the System Config menu shows only MTT subfields in the I/O BASE ADDRESS and MEM BASE ADDRESS fields. No other manufacturers are currently listed. The examples in Users Manual section 2.5.2 may show several different manufacturers, which do not apply in Nihon versions.

External clocking of individual sample conversions and external TTL-level trigger can be implemented using external inputs on the DSP4x40 clock board or the DSP4x30 digital I/O board. Software-initiated clocking of individual samples is also available. No hardware-based variable-level analog trigger exists.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C3x code.

Hardware Ratings and Software Throughput Specifications

The specifications below assume a system consisting of a DSP4x10 analog input board, DSP4x20 analog output board, and a DSP4x40 clock board.

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C30, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	100 kHz
1 channel Hard Disk Playback:	100 kHz
Max. 2-chn. Spectrum Analyzer Rate:	130 kHz
1024 pt FFT Accel. Disk File Rate:	17 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	UA
Maximum Ext. Mem. Config:	UA
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes, external connector
Host Data Transfer Method:	Memory-Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	8, 12-bits
Codec channels:	none
Expandable:	UA
Simultaneous S/H	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±10V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, if DSP4x50 filter board used
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	1/1200 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	8, 14-bits
Codec channels:	none
Expandable:	UA
Simultaneous S/H:	yes
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±10V
Output Impedance:	UA
Reconstruction Filters:	yes, if DSP4x50 clock board used
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	yes
Max. Sampling Rate:	200 kHz
Min. Sampling Rate:	1/1200 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The DSP4200 system board is shipped standard with 128k x 32 SRAM of combined program and data memory. Hypersignal uses less than 8k x 32 of for program storage at any one time. The remaining amount is used for processing and analog I/O data buffers. The large amount of data memory means that all board-related functions in Hypersignal are active and operate with maximum speed and size, including the Spectrum Analyzer, which can perform 2-channel 4096 pt. FFTs. Additional benefit from a memory expansion on the DSP4200 system would be provided mainly to the Stimulus & Response measurement function, which would allow larger stimulus and response sizes. Note that TMS320C3x devices suffer performance decrease inversely to the number of wait-states required by external SRAM access. Initial DSP4200 system memory configuration is zero wait-state; it is recommended that any memory expansion also utilize zero wait-states.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the MTT Instrumentation DSP4200 board in the Analog Conversion function (i.e. entry of ‘EXT’ for TRIGGER field is ignored).

3.14 Motorola

3.14.1 Motorola ADS56000 Development System (DSP56001)

Basic Descriptive and Installation Details

Use a hardware designator of 'ADS56-A' or 'ADS56-B' (ADC56000 or ADC16 analog I/O boards, respectively; see discussion below) in the Hypersignal software System Configuration menu. Use a hardware designator of 'ADS56' if the ADS56000 is to be used only as a "DSP Accelerator" (no analog I/O hardware connected). It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The Motorola ADS56000 system contains the Motorola 24-bit fixed-point DSP56001 device. It can provide analog I/O when connected to other external boards. The ADS56000 system is an external board that is interface to the PC using a 1/4 length 8-bit ISA bus XT-style connector.

Normally, the Motorola ADS56000 board is interfaced to the host PC through what is called the "ADM" card, a small interface board made by Motorola and plugged into the host PC, which is then connected to the ADS56000 board via 37-pin D-type connectors and 38-pin ribbon cable (ADS56000 board connector J1). When used with Hypersignal-Macro, Hypersignal-Macro EX, or Hypersignal-Acoustic software, the ADS56000 board requires a second host-PC connection through the Signalogic "ISAHI56" card, or "ISA bus host-interface for DSP5600x" card. This board is small, simple, and inexpensive. Like the ADM card, it also plugs into the host PC and connects directly to the ADS56000 board, except to connector J4, which is the DSP56001 host interface.

The ISAHI56 card is necessary to provide a fast method of interfacing to the DSP56001 in real-time, as the standard Motorola ADM card must communicate through onboard ADS56000 firmware which allows only a multiple-byte command and data packet protocol with polled handshaking, and is inadequate for continuous sampling rates above a few kHz. In order to operate correctly with the ADS56000 board, Hypersignal requires *both* the ADM and ISAHI56 cards to be installed and set to the same base address. At no time does installation of the ISAHI56 card cause interference with Motorola development utilities that depend on the standard ADM board to talk with the ADS56000.

NOTE: In all cases of Motorola ADS56000 board operation, Hypersignal expects the ADS56000 to be set up such that X: and Y: memory are the same. Examples are 4k x 24 P:, 2k x 24 X:, 2k x 24 Y: for standard ADS56000 boards, and 16k x 24 P:, 8k x 24 X:, 8k x 24 Y: for boards with the 32k x 8 RAM option. If you are using the 32k x 8 RAM option, be sure to install Motorola's new PAL, and also double-check with us or Motorola about the jumper settings, as previous Mototola ADS56000 manuals have had errors in the jumper lists.

Initial default Hypersignal I/O base address for the Motorola ADS56000 system is 100H (for *both* the ADM and ISAHI56 cards). On PC-XT machines 100H may not be a valid address (some hard disk controllers use this I/O space). If you get an error message of the form "ADM hardware not responding..." (Motorola ADM board error) or "ADS hardware not responding..." (ISAHI56 board error) or your system hangs up when you attempt to run functions requiring the ADS hardware, then check your system manual for possible hard disk controller hardware address conflicts. 300H may be a better address value in all cases. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address. The I/O base address jumper blocks function identically on both the Motorola ADM board (labeled as JG1) and ISAHI56 board (labelled as JP1); both are oriented identically, have two jumper positions, with four possible settings, as shown below:

address (top of board ^)	invalid	100H	200H	300H
jumper 1-2	in	out	in	out
jumper 3-4	in	in	out	out

Be sure that the I/O BASE ADDRESS field for "ADS" in the Hypersignal System Configuration menu matches the actual hardware jumper setting!

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP56001 code.

For Motorola ADS56000 hardware, the analog interface is totally dependent—in terms of channels, resolution, and anti-alias filtering—on what is connected to the "Eurocard" (or VMEbus) connector J3 or the Molex strip connector J5 on the ADS56000 board. Two such external analog boards are currently supported off-the-shelf by Hypersignal software: the Ariel ADC56000, and the Motorola DSP56ADC16 evaluation board. Other analog hardware can be interfaced and made operational through Hypersignal by modifying the DSP56001 source code—see section 4.3, Motorola DSP56001 DSP Source Code Interface, below. Without any external analog I/O hardware connected, the ADS56000 system can still be used as a "DSP Accelerator" for non-real-time file-based acceleration by the Hypersignal software.

Ariel ADC56000 Analog I/O Board

The Ariel ADC56000 has two 16-bit input and output channels (maximum sampling rate 70 kHz per channel). The Ariel ADC56000 also has either 10 kHz or 20 kHz optional low-order anti-aliasing input and smoothing output filters that can be enabled/disabled by onboard jumpers. The presence, and cutoff frequencies, of these filters must be specified when purchasing the ADC56000.

NOTE: On first revision ADC56000 boards the "INPUT 1" and "INPUT 2" labels are reversed on the silkscreen—simply connect the desired channel 1 input to the "INPUT 2" connector (same is true for "OUTPUT 1" and "OUTPUT 2"). Also, the D/A analog outputs on the ADC56000 drift away from zero when not used for a period of time; this causes output on the ADC56000 to begin with a "pop" as the D/A outputs are yanked back to normal signal levels. This "pop" becomes less noticeable as output events are repeated with less delay between them. This drift can be corrected with the following hardware change: install 1 Mohm resistors in parallel with capacitors C20 and C21 (one resistor each).

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Motorola ADC16 Analog I/O Board

The Motorola ADC16 evaluation board has one 16-bit input and output channel (maximum input sampling rate 100 kHz per channel at 16-bits and 400 kHz at 12-bits, maximum output sampling rate of 200 kHz). The Motorola ADC16 normally has no need for anti-aliasing input filters because of the inherent oversampling and digital filtering techniques used in the DSP56ADC16 sigma-delta input converter. A 9th-order elliptic lowpass filter is present on the board as a D/A output reconstruction filter, with a cutoff fixed at 20 kHz (not programmable, but bypassable using jumper).

NOTE: In its factory configuration, the ADC16 board is set for a 48 kHz sampling rate. It is not possible to control the ADC16 sampling rate via Hypersignal software—you must use the jumpers and/or external clock inputs on the ADC16 board to change sampling rates. You should enter a matching value in the software's SAMPLING FREQUENCY menu fields in order to ensure correct time unit display. The default value to enter would be 48 kHz.

Hardware Ratings and Software Throughput Specifications

NOTE: In the specifications below, values for the Ariel ADC56000 and the Motorola ADC16 evaluation board are separated (when different) by a "/".

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP56001, 97.7 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	105 kHz, MM
1 channel Hard Disk Playback:	105 kHz, MM
Max. 2-chn. Spectrum Analyzer Rate:	40 kHz/50 kHz
1024 pt FFT Accel. Disk File Rate:	13.5 frames/sec
Spectrum Analyzer Noise Floor:	85/91 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	4k x 24 P, 2k x 24 X, 2k x 24 Y
Maximum Ext. Mem. Config:	16k x 24 P, 8k x 24 X, 8k x 24 Y
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bit/1, 16-bit or 1, 12-bit
Codec channels:	none
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	85 dB/92 dB
THD:	UA
Voltage Range:	±3V to ±10V, TA / ±4V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, RP, Cheby5, RP, JB/SD
Programmable Sampling Rate:	yes/no (fixed at 48 kHz)
External Sample Clock:	yes/yes
Max. Sampling Rate:	70 kHz/100 kHz 16-bit, 400 kHz 12-bit
Min. Sampling Rate:	9.54 Hz/.01 Hz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bit/1, 16-bit
Codec channels:	none
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	85 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby5, RP, JB/ellip9, FC20, JB
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes/no (fixed at 48 kHz)
External Sample Clock:	yes/yes
Max. Sampling Rate:	140 kHz/200 kHz
Min. Sampling Rate:	9.54 Hz/.01 Hz
External TTL-Level Trigger:	no

Onboard Memory Usage

The ADS56000 development system is shipped standard with 4k x 24 of program memory, and 2k x 24 of X: data and Y: data memory. Maximum continuous sampling rates to hard disk are heavily dependent on the amount of X: and Y: data memory on the ADS board. ADS boards with the standard MCM6164 8k x 8 RAM devices (corresponding to the 2k x 24 X: and Y: amount) are capable of about 20 kHz continuous operation to/from hard disk; boards with MCM61256 32k x 8 devices (8k x 24 X: and Y:) are capable of about 95 kHz operation (see specifications below for exact figures). Also, with the 8k x 8 devices, the Spectrum Analyzer function is only capable of single-channel 256-pt. FFT operation. With either memory size, convolution/correlation is not enabled, and stimulus sizes for stimulus & response operation are limited.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with either the ADC56000 or the ADC16 in the Analog Conversion function (i.e. entry of ‘EXT’ for TRIGGER MODE, LEVEL field is ignored).

Sampling rates for the ADC16 evaluation board can be entered in the high-level software but have no effect, since the board is fixed to a 48 kHz rate. You can provide an external clock input to the ADC16 board, in which case you should enter a matching number in the software. You could also modify the hardware directly to allow use of onchip DSP56001 timers to generate a

clock rate, in which case you might also want to modify the Hypersignal DSP source code for the ADS system to utilize the software-driven programmble value written to shared memory by the high-level Hypersignal host software.

3.15 National Instruments

3.15.1 National AT-DSP2200 (DSP32C)

Basic Descriptive and Installation Details

Use a hardware designator of 'NAT32C' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The AT-DSP2200 contains the 32-bit floating-point AT&T DSP32C DSP chip, along with two 16-bit analog I/O channels. It is a full-length board with 16-bit ISA bus AT-style connector.

Analog input and output uses sigma-delta A/D and delta-sigma D/A converters, which provide programmable sampling rates, built-in anti-aliasing and reconstruction filters that track exactly with the programmed sampling rates, and linear phase. The Crystal Semiconductor CS5336 stereo audio codec and CS4328 stereo audio codec devices are used for A/D and D/A conversion, respectively. Both of these devices are rated at greater than 90 dB of dynamic range. The software-selectable sampling rates on this board must be one of 16 discrete values in a range from 4 to 51.2 kHz allowed by combinations of onboard crystal oscillators. Values such as 8, 16, 19.2, 24, 32, and 48 kHz, as well as popular audio rates such as 22.05 and 44.1 kHz, are possible. Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP32C Source Code Interface, it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion. Also, sampling rate oscillators on the AT-DSP2200 board can be replaced with a different value to achieve desired real-time sampling rates with optimum performance (no extra signal processing overhead).

Initial default Hypersignal I/O base address for the AT-DSP2200 board is 140H, which is the National factory default. To change the I/O base address, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal (with an ASCII text-editor). See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the AT-DSP2200 board I/O base address in these files. See Users Manual section 2.5.2, Initializing the System Configuration Menu, for general information on changing base I/O addresses.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock or non-standard crystal oscillators and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage. If you change oscillator values, then you will need to make minor modifications to the DSP32C source code in order to program the sampling rate latch correctly (because the built-in Hypersignal programming is disabled by the '-EXT' selection). See section 4.4, AT&T WE DSP32C DSP Source Code Interface.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32C code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
2 channel Hard Disk Record:	51.2 kHz
2 channel Hard Disk Playback:	51.2 kHz
Max. 2-chn. Spectrum Analyzer Rate:	51.2 kHz
1024 pt FFT Accel. Disk File Rate:	19 frames/sec
Spectrum Analyzer Noise Floor:	90 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	64k x 32
Maximum Ext. Mem. Config:	384k x 32
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	UA
Voltage Range:	±3V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P16
External Sample Clock:	yes
Max. Sampling Rate:	51.2 kHz
Min. Sampling Rate:	4 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, DS
Sin(x)/x Filters:	yes, DS
Programmable Sampling Rate:	yes, P16
External Sample Clock:	yes
Max. Sampling Rate:	51.2 kHz 16-bit
Min. Sampling Rate:	4 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The standard National AT-DSP2200 board contains 64k x 32 of zero wait-state external A area SRAM. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog I/O data buffers, file-based math/DSP function processing (acceleration), instrumentation and measurement calculations, and stimulus storage. The AT-DSP2200 SRAM can be expanded to 384k x 32 of total zero wait-state external A memory. We strongly suggest the 384k x 32 amount in order to fully enable all Hypersignal acceleration functions, allow continuous real-time functions to operate at full speed and capacity, and enable long stimulus & response operation.

The large amount of SRAM for reasonable cost makes this board an ideal candidate for use with the Real-Time Convolver, which allows real-time convolutions of acoustic/audio data with very long impulse responses (filters). The Real-Time Convolver allows mono or stereo convolutions at rates up to 44.1 kHz, and filters up to 50,000 pts.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the AT-DSP2200 board in the Analog Conversion function (i.e. entry of ‘EXT’ for TRIGGER MODE, LEVEL field is ignored).

3.16 NEC

3.16.1 NEC DDK-77P220 Development Board (77220)

Basic Descriptive and Installation Details

Use a hardware designator of 'NEC220' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The DDK-77P220 is the NEC-manufactured development board for the 24-bit fixed-point 77220 DSP chip. The 77P220 is the EPROM version of the 77220, with several pre-programmed routines and space for user-modifications, if necessary. The DDK-77P220 board contains a 14-bit μ-Law codec, which allows telecomm-quality analog I/O with fully adequate anti-aliasing and reconstruction filters. It is a full-length board with 8-bit ISA bus XT-style connector.

The DDK-77220 board uses both memory-mapped and I/O-mapped addresses in the host PC address space in order to communicate with Hypersignal software. The default memory base address is D800H, and can be entered directly in the System Configuration menu (MEM BASE subfield labeled "NEC"). The default I/O base address is 300H, and can only be changed by editing the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly—see the file "CONFIG.DOC" for descriptions and the sequence of all variables stored in the configuration file.

NOTE: You must ensure that the memory and I/O base addresses used in the System Configuration menu match those set by jumpers on the DDK-77220 board. The memory base address is set on the DDK-77220 board by jumpers J29 through J33. The current memory base address can be "read", left to right (or msb first) from jumpers J29, J30, J32, J31, and J33 in one's complement form (i.e. default on board should physically appear as :::\|:). Note that these jumpers represent only the five msbs of the memory address (i.e. 11011 in D800H); note also that J32 and J31 are not physically sequential. The I/O base address is set by jumpers J27 and J28, which represent the upper 2 bits of a 10-bit I/O address, the lower 8 bits of which are assumed to be zero. Thus, the default of J27 and J28 both set to the "left" would specify an I/O base address of 300H; J27 set to "left" and J28 set to "right" would specify an I/O base address of 200H. These are the only two combinations that should be used, since an address of 000H is not allowed by PCs, and an address of 100H may, in some machines, be used by the hard disk controller.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -8191 to 8191 for 14-bit A/D) and applied digitally, via 77220 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	77220, 122 nsec
Data Format, Wordlength:	fixed-point, 24-bit
1 channel Hard Disk Record:	8 kHz
1 channel Hard Disk Playback:	8 kHz
Max. 2-chn. Spectrum Analyzer Rate:	8 kHz
1024 pt FFT Accel. Disk File Rate:	UA
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	4k x 32; 4k x 24 data-only
Maximum Ext. Mem. Config:	4k x 32; 4k x 24 data-only
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	no
External Parallel Interface:	no
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	none
Codec channels:	1, 14-bit PD-9516AD μ-Law
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±2.5V
Input Impedance:	UA
Anti-Aliasing Filters:	yes, codec
Programmable Sampling Rate:	no
External Sample Clock:	no
Max. Sampling Rate:	8 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	none
Codec channels:	1, 14-bit PD-9516AD μ-Law
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	UA
THD:	UA
Voltage Range:	±2.5V
Output Impedance:	UA
Reconstruction Filters:	yes, codec
Sin(x)/x Filters:	yes, codec
Programmable Sampling Rate:	no
External Sample Clock:	no
Max. Sampling Rate:	8 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The DDK-77P220 board provides 4k x 32 of program memory and 4k x 24 of data memory. This limits the maximum file-based FFT acceleration and Spectrum Analyzer FFT sizes to 1024 pt, disables convolution/correlation acceleration, and limits stimulus size in stimulus & response operation. A memory expansion is not possible. Continuous disk transfer rates are limited also, but this has no impact since the maximum analog I/O sampling rate is only 8 kHz (codec).

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the NEC DDK77P220 in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

FFT sizes can only be 256 pt, 512 pt, or 1024 pt. Any other menu choices default to the nearest of these three values.

3.17 Signalogic

3.17.1 Signalogic Sig32C-4 (DSP32C)

1) When using the Sig32C-4, there are 4 programmable gain values possible for each channel (channel-independent). The following list is in dB:

0, 6.02, 12.04, 18.06

These values are achieved using software-programmable Burr-Brown amplifier stages on the board.

2) You must have legitimate signals connected between the V+ and V- inputs to see correct gain operation. Measurements on unconnected channels (ambient noise) or between single-ended inputs (V+ and GND) will not show correct gain values.

3) Zero gain full-scale input voltage for the Sig32C-4 is 1.4. Thus, entering '1.4' or 'v1.4' will show no gain, 'v0.14' would be 20 dB, '.014' would be 40 dB, etc.

4) See “GENERAL HARDWARE SUPPORT NOTES”, above for details and examples of gain list entry into instrumentation and Analog Conversion menus.

5) Default Sig32C-4 I/O base address is 290H. If this needs to be changed, it must be done in the HSMACRO.CNF or HSACOUS.CNF files (see config.txt file forlayout of these files).

6) The onboard clock frequency used to generate audio rates is 45.158 Hz. Hypersignal and DSPower software use an internal value of 45.1584 MHz, in order to produce “clean” Fs values without trailing digits. For example, if you select a sampling rate of 44.1 kHz, the software will indicate an exact match, but in reality your sampling rate will be 44.099609 kHz. In all cases, such error is less than .01 percent.

7) Connector Pinouts. On the backplane DB37 connector, pins 20-37 are ground. Other pins are as follows:

1	CH1+	ch 1 + differential input
2	CH1-	ch 1 - differential input
3	CH3+	ch 3 + differential input
4	CH3-	ch 3 - differential input
5	CH0+	ch 0 + differential input
6	CH0-	ch 0 - differential input
7	CH2+	ch 2 + differential input
8	CH2-	ch 2 - differential input
9	GND	analog ground
10	CH3	ch 3 analog output
11	CH1	ch 1 analog output
12	GND	analog ground
13	CH2	ch 2 analog output
14	CH0	ch 0 analog output
15	GND	analog ground
16	L/R	sample clock output
17	IRQ2	external interrupt input
18	EXT	external clock input
19	VCC	+5V output

Note that single-ended input is possible by tying negative differential inputs to analog ground. External clock input should be 512x the desired samplng rate. The actual sampling rate clock is available on the L/R output.

Pinout for the digital I/O connector, which is J4 dual inline header, is

as follows:

1	GND	digital ground
2	D15	data bit 15
.	.	
.	.	
9	D8	data bit 8
10	EN8	high-byte direction control
11	GND	digital ground

12	D7	data bit 7
.	.	
.	.	
19	D0	data bit 0
20	EN0	low-byte direction control
21	GND	digital ground
22	PWR	write strobe output
23	PRD	read strobe output
24	IRQ2	external interrupt input
25	GND	digital ground
26	VCC	+5V output

Using the EN0 and EN8 lines, digital I/O can be configured as 8-bits in, 8-bits out (or vice-versa), all in, or all out. Tying the EN lines low causes data bits to act as outputs; by default both EN lines are pulled up so all data bits are configured as inputs. Note that it is possible to connect the PWR strobe to EN lines to allow bidirectional I/O operation. Please refers to Sig32C System Users Manual for more information.

3.17.2 Signalogic Sig32C-8 (DSP32C)

Basic Descriptive and Installation Details

Use a hardware designator of 'SIG32C' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The Sig32C-8 contains the 32-bit floating-point AT&T DSP32C DSP chip, along with eight (8) 16-bit analog I/O channels. It is a 2/3 length board with 16-bit ISA bus AT-style connector.

Analog input and output uses sigma-delta A/D and delta-sigma D/A converters, which provide programmable sampling rates, built-in anti-aliasing and reconstruction filters that track exactly with the programmed sampling rates, and linear phase. Four (4) Crystal Semiconductor CS4216 stereo audio codec devices are used for A/D and D/A conversion. These devices are rated at around 80 dB of dynamic range. The software-selectable sampling rates on this board must be one of 16 discrete values in a range from 7.5 to 48 kHz allowed by combinations of two onboard crystal oscillators. Values such as 8, 16, 24, 32, and 48 kHz, as well as popular audio rates such as 22.05 and 44.1 kHz, are possible. Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP32C Source Code Interface, it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion. Also, sampling rate oscillators on the Sig32C-8 board can be replaced with a different value to achieve desired real-time sampling rates with optimum performance (no extra signal processing overhead).

The multichannel capabilities of the Sig32C-8 are impressive. All 8 input channels have independently software-selectable hardware gain amplifiers, which allow gain values from 0 to about 40 dB to be entered in Hypersignal instrument menus. Any two of the 8 channels can be used in the Hypersignal instrument functions (including Spectrum Analyzer, Digital Oscilloscope, and Stimulus & Response). Any combination of the 8 channels can be used for continuous disk signal record and generation. Although the specifications below list only 2-channel hard disk record and playback rates, it may be possible to record all 8 channels at rates above 16 kHz, depending on the host system. A 66 MHz 486 system, with 10 MHz ISA bus, and fast hard disk controller located on a local VESA or PCI bus, may be able to achieve 8-channel recording at up to 24 kHz per channel.

Additionally, the Sig32C-8 offers the QWbus interboard communication interface, which utilizes FIFOs to allow both peer-to-peer data transfer and broadcast transfer. The QWbus also allows up to 16 lines of TTL-level external parallel digital

I/O. There is also available a Sig32C board with the analog I/O section replaced with a prototyping area, to which the DSP32C serial port signals have been brought out to adjacent headers. For fully detailed information about the Sig32C-8, see the Sig32C System Users Manual.

Initial default Hypersignal I/O base address for the Sig32C-8 board is 290H, which is the Signalogic default. To change the I/O base address, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal (with an ASCII text-editor). See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the Sig32C-8 board I/O base address in these files. See Users Manual section 2.5.2, Initializing the System Configuration Menu, for general information on changing base I/O addresses.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock or non-standard crystal oscillators and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage. If you change oscillator values, then you will need to make minor modifications to the DSP32C source code in order to program the sampling rate latch correctly (because the built-in Hypersignal programming is disabled by the '-EXT' selection). See section 4.4, AT&T WE DSP32C DSP Source Code Interface.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP32C code.

1) When using the Sig32C-8, there are 16 programmable gain values possible for each channel (channel-independent). The following list is in dB:

0, 7.5, 15, 20, 22.5, 27.5, 35, 40, 42.5, 47.5, 55, 60, 62.5, 67.5, 75, 82.5

The values 20, 40, and 60 are achieved using only the programmable Burr-Brown amplifier stages on the board. The values 7.5, 15, and 22.5 are achieved using only the internal gain stages inside the CS4216 codec A/D converters. Note that for version 3 boards, any values that use the Burr-Brown amps (anything above 22.5 dB) can only be set in "channel pairs" —channel 0 and 1 together, channel 2 and 3, etc. For version 4 boards, the gain can be set independently on all eight channels, regardless of value.

2) You must have legitimate signals connected between the V+ and V- inputs to see correct gain operation. Measurements on unconnected channels (ambient noise) or between single-ended inputs (V+ and GND) will not show correct gain values.

3) Zero gain full-scale input voltage for the Sig32C-8 is 1.4. Thus, entering '1.4' or 'v1.4' will show no gain, 'v0.14' would be 20 dB, '.014' would be 40 dB, etc.

4) See "GENERAL HARDWARE SUPPORT NOTES", above for details and examples of gain list entry into instrumentation and Analog Conversion menus.

5) Default Sig32C-8 I/O base address is 290H. If this needs to be changed, it must be done in the HSMACRO.CNF or HSACOUS.CNF files (see config.txt file for layout of these files).

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP32C, 80 nsec
Data Format, Wordlength:	floating-point, 32-bit
2 channel Hard Disk Record:	48 kHz
2 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	48 kHz
1024 pt FFT Accel. Disk File Rate:	19 frames/sec
Spectrum Analyzer Noise Floor:	80 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	32k x 32
Maximum Ext. Mem. Config:	128k x 32
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	80 dB
THD:	UA
Voltage Range:	±1.4V
Input Impedance:	10k || 12 pF, RP
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P16
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	7.5 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	80 dB
THD:	UA
Voltage Range:	±2.98V
Output Impedance:	UA
Reconstruction Filters:	yes, DS
Sin(x)/x Filters:	yes, DS
Programmable Sampling Rate:	yes, P16
External Sample Clock:	yes
Max. Sampling Rate:	48 kHz 16-bit
Min. Sampling Rate:	7.5 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The standard Sig32C-8 board contains 32k x 32 of zero wait-state external A area SRAM. Hypersignal uses less than 4k x 32 of this memory for program storage, and the remainder for analog I/O data buffers, file-based math/DSP function processing (acceleration), instrumentation and measurement calculations, and stimulus storage. The Sig32C-8 SRAM can be expanded to 128k x 32 of total zero wait-state external A memory. We strongly suggest the 128k x 32 amount in order to fully enable all Hypersignal acceleration functions, allow continuous real-time functions to operate at full speed and capacity, and enable relatively long stimulus & response operation.

The medium amount of SRAM for reasonable cost makes this board a good candidate for use with the Real-Time Convolver, which allows real-time convolutions of acoustic/audio data with very long impulse responses (filters). Mono or stereo convolution with filter lengths up to 10,000 pts. and rates up to 44.1 kHz are possible. The drawback in this situation is the noise-floor performance of the board, which is about 80 dB.

Additional Notes About Hypersignal Software Operation

1) Currently, external triggering operation is inactive with the Sig32C-8 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER MODE, LEVEL field is ignored).

2) In both the instrument menus and the Analog Conversion menu, programmable gain values can be entered as either full-scale voltage values, or as dB gain values. Prefixing a value with 'd' specifies dB; prefixing a value with 'v' or no prefix specifies full-scale voltage (default mode). For example, entry of

d20,d40,2.8

in the I/O GAIN LIST field in the Analog Conversion menu would specify 20 dB of gain an trace 1 (first physical channel in CHANNEL LIST field), 40 dB of gain on trace 2 (second physical channel in CHANNEL LIST field), and a full-scale voltage

range of 2.8 on trace 3. Remember, when entering full-scale voltage ranges, you enter a smaller value to get increased gain. For example, since the Sig32C-8 has a 1.4V full-scale input (zero gain), an entry of '.14' would specify a gain of 10.

3) From the Spectrum Analyzer and Digital Oscilloscope function menus, the INPUT RANGE field can be a list of values. For example, entry of

5,10

would be 5 Volts full-scale for trace 1, and 10 Volts full-scale for trace 2. An entry of

d7.5,d10

would be 7.5 dB of gain on trace 1, and 10 dB of gain on trace 2.

4) In the Analog Conversion menu, the I/O GAIN LIST field can be a list of values, separated by commas. Use a '|' to separate input from output values, with output values following the '|'. Note that output values can either be ONE software scaling value or a LIST of hardware attenuation values. If any entered output value is prefaced by 'v' or 'd' (as described above), then hardware attenuation is assumed (attenuation values in dB should NOT be entered as negative). Otherwise, a single value is assumed as a software multiplier.

5) When using the Sig32C-8, there are 16 programmable gain values possible. The following list is in dB:

0, 7.5, 15, 20, 22.5, 27.5, 35, 40, 42.5, 47.5, 55, 60, 62.5, 67.5, 75, 82.5

The values 20, 40, and 60 are achieved using only the programmable Burr-Brown amplifier stages on the board. The values 7.5, 15, and 22.5 are achieved using only the internal gain stages inside the CS4216 codec A/D converters. Note that for v3 boards, any values that use the Burr-Brown amps (anything above 22.5 dB) can only be set in "channel pairs" —channel 0 and 1 together, channel 2 and 3, etc. For v4 and higher boards, the gain can be set independently on all eight channels, regardless of value. There are other gain values that can possibly be set, but you would need to make simple modifications to the DSP32C Source Code Interface routines to do this (see section 4.4, AT&T WE DSP32C DSP Source Code Interface).

6) You must have legitimate signals connected between the V+ and V- inputs to see correct gain operation. Measurements on unconnected channels (ambient noise) or between single-ended inputs (V+ and GND) will not show correct gain values.

7) Zero gain full-scale input voltage for the Sig32C-8 is 1.4. Thus, entering '1.4' or 'v1.4' will show no gain, 'v0.14' would be 20 dB, '.014' would be 40 dB, etc.

8) Output channels higher than 1 (2-7) are *not* available during stimulus and response operation. Input can be looped back on all channels in monitor mode, but no stimulus data will be output during the actual measurement.

3.17.3 Signalogic SigBook (DSP5630x)

Connecting SigBook to PC

1. Make sure the power to PC is turned off.

2. Plug the DC power input from power adaptor into the SigBook. Plug the DC adaptor into the AC outlet. Red LED at front of the SigBook case should be on. If red LED is not on, disconnect power adapter immediately and contact Signalogic.

Note: The power adaptor included by Signalogic is 7-9 Vdc. Power adaptors outside this range or AC-to-AC should not be used.

3. Connect female side of your parallel port cable into the SigBook and the male side into PC parallel port.

Note: If you do not have a free parallel port available for SigBook (for example, because the printer or ZIP drive already occupies it) then you have to install additional parallel port card. Additional parallel port must have different base address assigned. (On most PCs, base address 378H is reserved for parallel port 1, 278H for parallel port 2 and 268H for parallel port 3).

4. Turn on the PC. Make sure parallel port is set to EPP mode. It may be necessary to run the BIOS setup to check parallel port mode and base address. Most PCs allow BIOS set-up to be run by pushing <DEL> or <F1> keys during PC boot.

Note: SigBook will not work if parallel port is set to "compatible mode" or "bidirectional mode". On newer PCs, the parallel I/O controller is implemented on the motherboard and you can set the parallel port mode from BIOS setup under the "peripheral

configuration" menu item. If the parallel I/O controller is on separate card, check your I/O card manual for instructions about how to set EPP mode.

SigBook Testing

1. Run Windows

2. Install DSPower and/or Hypersignal software, if not already installed.

3. Select HwLib Digital Scope Demo by double clicking this icon from Signalogic DSP Software program group (Win 3.1x) or by clicking icon from the Signalogic DSP Software folder (Win 9x).

4. Select SigBook from the list of devices.

5. Make sure the Base I/O Address field (shown below list of devices) corresponds to the parallel port your SigBook is connected to. See "Connecting SigBook to PC" for more information.

Note: Default base address for parallel port 1 (LPT1) is 378H. If parallel port 1 is used then you may have to use parallel port 2 (LPT2), and the base address should be 278H.

6. Click on th OK button. DScope window should appear. Click several times on "vertical scale expand" button (top-left of toolbar). If a noise signal appears on the window your SigBook hardware and software is correctly installed and operating.

Remark: The "Mic In" input on the SigBook unit is currently not availble. Do not use.

3.17.4 Signalogic SigC31-4 (TMS320C31)

1) Currently, Hypersignal drivers are not working correctly with the DMA transfer mode option on the SigC31-4 board. This means the System Config menu selection of 'SIGC31-B' should be used in all cases.

2) When using the SigC31-4, there are 4 programmable input gain values possible. The following list is in dB:

0, 6.02, 12.04, 18.06

The above values are achieved using the programmable Burr-Brown PGA205 amplifier devices on the board. Gain can be set on each input channel independently.

3) You must have legitimate signals connected between the V+ and V- inputs to see correct gain operation. Measurements on unconnected channels (ambient noise) or between single-ended inputs (V+ and GND) may not show correct gain values.

4) Zero gain full-scale input voltage for the SigC31-4 is 3.68. Thus, entering '3.68' or 'v3.68' in an instrument INPUT RANGE field or Analog Conversion GAIN LIST field will show no gain, 'v0.92' would show 12 dB of gain, '.46' would be 18 dB of gain, etc.

5) Gain values can be set as full-scale voltage ranges or in dB values. See "GENERAL HARDWARE SUPPORT NOTES", above for details and examples of gain list entry into instrumentation and Analog Conversion menus.

The ' -B' indicates no DMA tranger; omit this to allow DMA transfer.

6) Default SigC31-4 I/O base address is 290H. This value can be modified in the DSPower DSP/Analog Hardware Manager dialog box. If this needs to be changed, it must be done in the HSMACRO.CNF or HSACOUS.CNF files (see config.txt file for layout of these files).

Please refers to Signalogic SigC31 System Users Manual for more information.

3.17.5 Signalogic SigC32-2 (TMS320C32)

The documentation for the SigC32-2 board is similar to the SigC31-4 board. However, there are significant differences, mainly in the analog I/O subsection. The SigC31-4 Users Manual should be used in conjunction with the additional information below.

Note also that there should be a backplane connector pin-out included for the SigC32-2 board; the pin-out is NOT the same as for the SigC31-4 board.

1) The current SigC32 driver does not support onchip ('C32 chip) DMA transfers to host when used with analog I/O functions. Although non-analog I/O functions will operate correctly with either driver selection (DMA enabled or disabled), it is safer, and there is minimal performance impact, to always use the "no DMA" option. In DSPower software, this would be a selection like "SigC32-2, no DMA transfer" in the board selection area of the DSP/Analog Hardware Manager dialog box. When manually invoking Hypersignal software in user mode, you must specify the board type in the System Config menu. The first four (4) fields of this menu must specify the correct "board designator"; for example, you might enter:

ANALOG CONERSPN : SIGC32-B

DSP ACCELERATOR : SIGC32-B

DIGTAL SCOPE : SIGC32-B

SPECTRUM ANALYZER: SIGC32-B

2) The SigC32 generates its analog I/O sampling rate clock using on onchip ('C32 chip) timer 0 or timer 1 (controlled by jumper). Thus, sampling frequencies are based on processor clock. To ensure correct software readout, for example units on instrument or waveform display axes, be sure to enter the correct processor clock for your board. When using DSPower software, the processor clock should be entered (in MHz) in the Processor Clock field in the DSP/Analog Hardware Configuration dialog box; note that entries are in MHz, so values like 40, 50, or 60 are typical. When running Hypersignal software in user mode, you must specify the correct processor clock rate in the System Config menu as an attribute after the board designator (see item 1 above). For any or all of the first (4) fields, you might enter:

SIGC32-B-59.52

or

SIGC32-B-40

or

SIGC32-B-60

3) Full-scale input voltage for the SigC32-2 is 3.1625, and full-scale output voltage for the SigC32-2 board is 3.

4) Default SigC32-2 I/O base address is 290H. If this needs to be changed, it must be done in the HSMACRO.CNF or HSACOUS.CNF files (see config.txt file for layout of these files).

5) Connector Pinouts.

On the backplane DB37 connector (P1), pins are as follows

1 ch1 + differential output
2 ch1 - output
3 ch0 + differential output
4 ch0 output
5 ch0 + differential input
6 ch0 - differential input
7 ch1 + differential input
8 ch1 - differential input
9 not connected
10 analog ground
11 digital I/O 14
12 digital I/O 12
13 digital I/O 10
14 digital I/O 8
15 digital I/O 7
16 digital I/O 5
17 digital I/O 3
18 digital I/O 1
19 analog ground

20 ch1 - differential output
21 analog ground
22 ch0 - differential output
23 analog ground
24 analog ground
25 analog ground
26 analog ground
27 analog ground
28 analog ground
29 digital I/O 15
30 digital I/O 13
31 digital I/O 11
32 digital I/O 9
33 analog ground
34 digital I/O 6
35 digital I/O 4
36 digital I/O 2
37 digital I/O 0

Note that single-ended input is possible by tying negative differential inputs to analog ground and pins 23 - 28 , 10, 19, 33 can be used as analog ground connections.

Pinout for the P2 connector, which is J4 dual inline header, is as follows:

1 Pin 1 - digital ground
2 Pin 2 - external clock input (CLK+)
3 Pin 3 - external clock retun (CLK-)
4 Pin 4 - digital ground
5 Not connected
6 Not connectd
7 Pin 7 - analog ground
8 Pin 8 - ch1 - differential output
9 Pin 9 - ch1 + differential output
10 Pin 10 - analog ground
11 Pin 11 - analog ground
12 Pin 12 - ch0 - differential output
13 Pin 13 - ch0 + differential output
14 Pin 14 - analog ground

Using the XF0 and XF1 lines, digital I/O can be configured as 8-bits in, 8-bits out (or vice-versa), all in, or all out. Setting the XF flags low causes data bits to act as outputs; by default both XF flags are pulled up so all data bits are configured as inputs.

Please refers to Signalogic SigC31 System Users Manual for more infomation.

3.17.6 Signalogic SigC44-PC/104 (TMS320C44)

3.7.6.1 Introduction

This section an except from the hardware specification for the Signalogic TMS320C44 PC/104 board (SigC44-PC/104). The SigC44-PC/104 is a PC/104 format small footprint (about 3.5" x 3.5") board with Texas Instruments TMS320C44 DSP and dual-port RAM host PC interface. There is also additional digital I/O memory mapped onto the Global Bus (Global Bus+ Connector). TMS320C44 processor clock speeds can be 40, 50, or 60 MHz.

The board has the following key features:

· PC/104 format, very small form factor

· Texas Instruments TMS320C44 DSP; 40, 50, or 60 MHz floating-point DSP

· 128 or 512k x 32 zero wait-state external SRAM

· four (4) Communication Ports, or “Comm Ports” (each 8-bit digital I/O)

· JTAG 1149.1 Emulator Header

· 8K x 32 dual-port RAM (DPRAM) host-interface, including 8 DPRAM semaphores

· 32-bit parallel digital I/O Global Bus+ Connector

The PC/104 bus format is essentially PC ISA bus format in a very small footprint for embedded-PC systems. PC/104 bus signals are compatible with ISA bus signals and code development/test normally takes place in a PC with an ISA to PC/104 carrier (extender) board.

For further information on the PC/104 format and bus please refer to the PC/104 specification (currently at Version 2.3). This is available from the PC/104 Consortium website at www.controlled.com/pc104.

Below is a top-view of the SigC44, including orientation of Global Bus+ connector:

3.17.6.2 TMS320C44

The TMS320C44 is a high performance 32-bit floating-point DSP and has the following key features:

40, 50, or 60 MFLOPs (50, 40, or 33 nsec instruction cycle)

· four (4) communication (“Comm”) ports

· six (6) channel onchip DMA

· 32-bit data local bus

· 32-bit data global bus

· 24-bit address bus

· 2 k x 32 onchip memory

· 128 x 32 onchip instruction cache

· 304 pin PQFP footprint

A 50 / 60 MHz crystal oscillator driving the TMS320C44-50 / TMS320C44-60 provides 50 / 60 MFLOPs performance (40 / 33 nsec instruction cycle).

The JTAG header connects to the TMS320C44 emulator pins. This is a 14-pin (2x7) right-angled header (pin 6 removed) and is located on the edge of the board. This allows emulator-based hardware and software debug to take place with the board in the system. There are several combined hardware/software debugger packages available which allow high level software debug, using C source code and symbol information. These include White Mountain Vista-MP 510, CodeComposer for ‘C4x, and others.

The TMS320C44 is interfaced to the PC/104 bus by dual-port RAM. A Lattice Semiconductor ispLSI2064 EPLD is used to interface the TMS320C44 and DPRAM to the PC/104 bus.

3.17.6.2.1 TMS320C44 DPRAM

The dual-port RAM is 8K x 32 (2x IDT7025 or QS7025 8k x 16 DPRAM devices) on the TMS320C44 side and is located on the local bus at address 0x00000000. This is where the reset and interrupt vectors are located. The DPRAM operates at 1 wait-state. The RDY signal is generated by the hardware to allow for DPRAM BUSY arbitration.

The TMS320C44 operates in its microprocessor mode and executes code from DPRAM from address 0x00000000 (ROMEN = '0' and RESETLOC 0,1 = '00'). A "talker" program must be first loaded to allow host access to the full memory range of the TMS320C44.

The PC/104 host interface controls the TMS320C44 reset and interrupt signals.

3.17.6.2.2 TMS320C44 DPRAM Semaphores

The 8 dual-port RAM semaphores within the DPRAM devices are implemented. These use data bit 0 in host interface register at Base + 0xA addresss. See section 3.4, Accessing DPRAM Semaphores From Host.

3.17.6.2.3 TMS320C44 SRAM

The SRAM is either 128k x 32 or 512k x 32 (4x 128 or 512k x 8 SOJ devices) on the TMS320C44 local bus (STROBE1) at address 0x00800000. The SRAM devices operate at 0 wait-state. The RDY signal is grounded for 0 wait-state operation.

The SRAM devices should be 20 nsec access time for 40 MHz operation, 15 nsec for 50 MHz operation, and 12 nsec access time for 60 MHz operation. The SOJ packaging used by the SRAM devices has center VCC and GND pins. The 128k SRAM devices are 32-pin SOJ package and the 512k SRAM devices are 36-pin SOJ package.

3.17.6.4 TMS320C44 Memory Map

The TMS320C44 local bus memory map is as follows

Address Start	Address Stop	Strob	Memory Type	Wait States
0x00000000	0x00001FFF	Local Strobe 0	8 K DPRAM	1 ws
0x00002000	0x000FFFFF	Local Strobe 0	DPRAM Reflections	1 ws
0x00010000	0x001000FF -		C44 Pripheals	
0x00100100	0x002FF7FF		C44 Reserved -	
0x002FF800	0x002FFBFF		C44 RAM BLK 0-	
0x002FFC00	0x002FFFFF		C44 RAM BLK1	
0x00300000	0x003FFFFF	Local Strobe 0		7 ws (internal RDY)
0x00400000	0x00400007	Local Strobe 0	8 DPRAM Semaphores	1 ws
0x00400008	0x004FFFF	Local Strobe 0	DPRAM Sema Refl	1 ws
0x00500000	0x00800000	Local Strobe 0		7 ws (interal RDY)
0x00800000	0x0087FFFF	Local Strobe 1	128K/512K SRAM	0 ws
0x00880000	0x00FFFFF	Local Strobe 1	SRAM reflections	0 ws
0x01000000	0x7FFFFFF		Local bus aliases	

The TMS320C44 global bus memory map is as follows:

Address Start	Address Stop	Strobe	Memory Type	Wait States
0x80000000	0xFFFFFFFF	Global Strobe 0	Global Bus+ Access	RDY-dependent

0x80800000	0xFFFFFFFF	Global Strobe 0	Global Bus+ Access	0 ws

Note about global bus address above: bit 23 (A23) high causes auto-acknowledge to occur on Global Bus. This can be useful for simulating operation of peripheral hardware which is not connected and would otherwise cause a processor hang-up due to RDY (acknowledge) signal not being received by the processor.

The TMS320C44 operates in microprocessor mode. The ‘C4x DSP core supports 32 bit addressing, but only 24 address lines are pinned out on the ‘C44 package.

3.17.6.2.5 TMS320C44 Communication Ports

The TMS320C44 communication ports 1, 2, 4 and 5 are pinned out to high-density connectors. The connectors are Ariel pin compatible on a 0.050” pitch. Series termination resistors of 22R are incorporated into the signal lines, to maintain data integrity and also allow some protection against misuse. The communication port signals are unbuffered.

3.17.6.2.6 TMS320C44 External Interrupts

The following TMS320C44 external interrupts are supported:

C44 Interrupt	Type	Source
NMI*	Edge	PC Interface
IIOFO*	Edge	Global Bus+ Connector
IIOF1*	Edge	-
IIOF2*	Edge	PC Interface
IIOF3*	Edge	PC Interface

Please refer to SigC44-PC/104 hardware manual for more information.

3.18 Sonitech

3.18.1 Sonitech Spirit-30 (TMS320C30)

Basic Descriptive and Installation Details

For use as a DSP accelerator, and/or with an “IC-100" external analog I/O unit, enter a hardware designator of ‘SONC30’ in the Hypersignal software System Configuration menu. For use with an ”SAIB" external analog I/O unit, enter ‘SONC30-SAIB’ in the System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to ‘Y’) if you have a 80286, 80386, or 80486 machine.

The Spirit-30 contains the Texas Instruments TMS320C30 32-bit floating-point device. It is a full-length board with 16-bit ISA bus AT-style connector.

When used with an “IC-100" analog I/O external unit connected, becomes a two-channel system capable of 16-bit A/D and D/A. The IC-100 contains two 16-bit sigma-delta A/D converters that contain extremely high-quality anti-aliasing filters with exactly linear phase. The IC-100 also contains two traditional D/A converters with analog reconstruction lowpass filters set for a cut-off frequency of about 48 kHz. The IC-100 sample rate is fixed at 100 kHz; on a fast system (25 MHz 80386 with 28msec avg. access time hard drive) you should be able to continuously record/playback to/from hard disk one channel at 100 kHz or two channels to/from RAM disk. The Spectrum Analyzer and Digital Oscillocope functions should operate (either one channel or two channel), at 100 kHz. Currently, Hypersignal allows entry of sampling rates from 0 to 100 kHz (the particular sigma-delta A/D converters used in the IC-100 are limited to 100 kHz when operating in 16-bit mode), to allow for the possibility that the user may provide external clock or change the oscillator inside the IC-100.

3.18.1 Sonitech Spirit-30 (TMS320C30)

When used with an "SAIB" (stereo audio interface box) analog I/O external unit connected, the Spirit-30 becomes a two-channel system capable of 16-bit A/D and D/A at programmable sampling rates ranging from 5.5 kHz to 48 kHz, with sigma-delta anti-aliasing input filters and delta-sigma reconstruction output filters that automatically track with the programmed sampling rate, and provide exactly linear phase. The SAIB has many impressive features suitable for speech, telecomm, and audio applications, including:

1) programmable selection of input source (line-level, microphone, handset, telephone line; Hypersignal defaults to line-level)

2) programmable hardware-controlled input gain and output attenuation (Hypersignal defaults to unity gain and attenuation)

3) programmable data format (16-bit or 8-bit μ-Law/A-Law; Hypersignal defaults to 16-bit)

4) programmable headphone enable/disable (Hypersignal defaults to enabled)

5) programmable telephone line control and status functions (Hypersignal defaults to the telephone interface, which is an option for the SAIB that must be purchased additionally, disabled)

Currently, these functions cannot be controlled from Hypersignal menus, but must instead be set using two control words stored in the HSMACRO.CNF configuration file (see notes below).

The Spirit-30 board uses I/O-mapped addresses in the host PC address space in order to communicate with Hypersignal software. The default I/O base address is 300H, and can only be changed by editing the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly—see the file "CONFIG.DOC" for descriptions and the sequence of all variables stored in the configuration file.

Below are two general items of note for proper Spirit-30 operation with the Hypersignal software, and several specific notes about use of the Spirit-30 with either the IC-100 or SAIB.

1) *Early revision Spirit-30 boards may not operate correctly at host I/O bus rates above 8.33 MHz.* Version 3.11 of the Spirit-30 has been successfully tested at an 11 MHz I/O bus rate. Symptoms of an I/O bus speed problem include a lock-up, or freeze, intermittent "spikes" in the acquired data, and excessive jitter or "waver" in waveforms (which is easily detected in a Digital Oscilloscope display of a sine wave, with trigger enabled).

2) *The Spirit-30 must have an "IC-100" or "SAIB" external unit connected for use with Hypersignal analog I/O functions.* Without an analog unit connected, the Spirit-30 can still be activated in the DSP ACCELERATOR field in the System Configuration menu.

NOTES FOR IC-100 USE: Hypersignal expects to see the IC-100 connected to serial port 0 (top connector on Spirit-30 board, where top is furthest away from the host I/O bus connector). This allows IC-100 units (which are not typically shipped with external power supplies) to use the +5V power supplied on serial port 0 (serial port 1 does not provide power). To change Hypersignal operation with the IC-100 to serial port 1, the TMS320C3x DSP Source Code Interface (optional purchase; see section 4.5, Texas Instruments TMS320C3x DSP Source Code Interface) must be changed and reassembled/recompiled.

Connecting the IC-100 can be tricky. **For IC-100 units without keyed cables, connect one end to the IC-100 so that the red stripe is at the upper right of the connector (as you look at the back of the IC-100 unit in its upright position), and connect the other end to the *top* connector on the Spirit-30 board so that the red stripe is towards the *top* of the board, where "top" indicates the edge of the board away from the host I/O bus connector.**

NOTES FOR SAIB USE: **Hypersignal expects to see the SAIB connected to serial port 1 (bottom connector on Spirit-30 board, where bottom is closest to the host I/O bus connector). This allows IC-100 units (which are not typically shipped with external power supplies) to use serial port 0, which provides +5V (serial port 1 does not). To change Hypersignal SAIB operation to serial port 0, the TMS320C3x DSP Source Code Interface (optional purchase; see section 4.5, Texas Instruments TMS320C3x DSP Source Code Interface) must be changed and reassembled/recompiled. To change Sonitech software utilities to use serial port 1, these programs must be changed and reassembled/recompiled.**

There are two words in the HSMACRO.CNF file that can be used to control the SAIB: "CS42xx_CTRL_WORD" (corresponding to the SAIB documentation CMC0 word) and "CS42xx_DATA_WORD" (corresponding to the SAIB documentation DMC word). The default bit settings and position in the HSMACRO.CNF file for these values is documented in the CONFIG.DOC file, located on distribution diskette #2. Any change to these words from their default setting as listed in the CONFIG.DOC file will prevent Hypersignal from initializing the changed bits. For example, if you change the default sampling rate field (DRF) or the crystal selection (CSEL) bits then, Hypersignal will no longer program these fields, and instead merely pass them along to the Spirit-30 board.

Hypersignal default settings for the Crystal Semiconductor 4215 part inside the SAIB are as follows: DAA (telephone interface) disabled, speaker output amps enabled, output level at 2.88 Vpp, stereo operation, 16-bit data, loop-back (monitor) path and tests disabled, headphone enabled, line-level output and input enabled (handset and microphone input disabled), overrange detection disabled, and no input gain or output attenuation.

NOTES FOR SAIB USE, cont: **Hardware gain in the SAIB is not currently specifiable from menus in Hypersignal; however, the standard digital scaling (and offset), implemented by the TMS320C30 in Hypersignal analog I/O functions is active. Thus, the INPUT RANGE fields in the instrumentation functions should be set to 2.8, since this is the default setting (2.88 Vpp) for the SAIB control words used by Hypersignal. If you change this value, or change the hardware input gain settings, then you must set the INPUT RANGE field accordingly, in order to maintain accurate display readout. It is possible that at some future point Hypersignal will allow entry of SAIB hardware gain values, and set display scaling appropriately.**

The following 14 sampling rates can be set exactly when using the SAIB with Hypersignal (kHz): 5.5125, 6.615, 8, 9.6, 11.025, 16, 18.9, 22.050, 27.42875, 32, 33.075, 37.8, 44.1, and 48. All other entered rates quantize to the closest of those possible.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C3x code.

Hardware Ratings and Software Throughput Specifications

NOTE: **In the specifications below, values for the IC-100, and SAIB external analog I/O units, respectively, are separated (when different) by a "/".**

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C30, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	100 kHz
1 channel Hard Disk Playback:	100 kHz
Max. 2-chn. Spectrum Analyzer Rate:	100 kHz
1024 pt FFT Accel. Disk File Rate:	18 frames/sec
Spectrum Analyzer Noise Floor:	95 dB / 80 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	32k x 32
Maximum Ext. Mem. Config:	128k x 32
Ext. Sync Serial Interface:	yes
Ext Async. Serial Interface:	yes
External Parallel Interface:	yes
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits SD
Codec channels:	none / 2, 8-bit μ-Law or A-Law
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	92 dB / 80 dB
THD:	UA
Voltage Range:	±4V / ±2.8
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	no, (fixed at 100 kHz) / yes, P14
External Sample Clock:	yes / no
Max. Sampling Rate:	100 kHz / 48 kHz
Min. Sampling Rate:	0 / 5.5 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none / 2, 8-bit μ-Law or A-Law
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	90 dB / 80 dB
THD:	UA
Voltage Range:	±4V / ±2.8V
Output Impedance:	UA
Reconstruction Filters:	yes / yes, DS
Sin(x)/x Filters:	no / yes, DS
Programmable Sampling Rate:	no (fixed at 100 kHz) / yes, P14
External Sample Clock:	yes / no
Max. Sampling Rate:	100 kHz / 48 kHz
Min. Sampling Rate:	0 / 5.5 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The Spirit-30 is shipped standard with 32k x 32 of memory (either program or data). Hypersignal uses less than 8k x 32 of memory for program storage. A memory upgrade to a total of 128k x 32 of memory improves stimulus & response operation by providing for much larger stimuli, and enables convolution/correlation operation. All other functions should perform equally well in either case.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the Spirit-30 board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

The IC-100 is fixed at a 100 kHz sampling rate for both input and output operations and for either one or two channel operation. You could provide an external clock input to the IC-100, in which case you should enter a matching sampling rate in the software.

3.19 Techron

3.19.1 Techron TEF20HI System (DSP56001)

Basic Descriptive and Installation Details

> **NOTE: You must have an "HI" (High-Speed Interface) TEF20 system in order to operate the Hypersignal software with the TEF20. A standard TEF20 system (non-HI) will not work.**

Use a hardware designator of 'TEF20HI' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled. The TEF20 system option is only active with Hypersignal-Acoustic software.

The TEF20HI system contains the Motorola DSP56001 24-bit fixed-point DSP device. It is shipped standard with two 16-bit A/D input channels and one 16-bit D/A output channel. The TEF20 is an external system that interfaces to the PC using the 1/4 length HI (high-speed interface) card, which uses an 8-bit ISA bus XT-style connector.

All analog I/O uses a fixed sampling rate of 48 kHz Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP56001 Source Code Interface, it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion.

Two-channel or one-channel input operation is supported by all Hypersignal analog input functions, including Analog Conversion, Digital Oscilloscope, and Spectrum Analyzer. Input is performed with sigma-delta A/D converters. One-channel output operation is supported by D/A output from time domain and frequency domain displays, the Analog Conversion function, and the stimulus & response function. Output is performed using a standard successive-approximation D/A converter, and an 8-pole Chebyshev reconstruction filter with linear phase compensation. Two-channel input triggering combinations (e.g. using one channel solely for triggering, and the other for data) are supported. Input and output channels have fixed-frequency anti-aliasing filters installed.

Additionally, the TEF20HI system contains both line-level input and microphone input connectors. Either set of connectors can provide the input data at any one time—which one is active can be controlled from the Hypersignal menus. Also, the microphone inputs incorporate independently controlled programmable gain circuitry. Each input can be programmed independently with one of 16 levels of gain (from 0 to 60 dB in 4 dB increments) from Hypersignal instrument and continuous disk transfer menus. See the section titled "Additional Notes About Hypersignal Software Operation", below, for information on setting these parameters from Hypersignal menus.

Initial default Hypersignal I/O base address for the Techron TEF20HI system is 308H, which is also the Techron initial factory default. The current value should be entered in the "ADS" subfield of the I/O BASE ADDRESS field in the System Configuration Menu. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the I/O BASE ADDRESS field, for details on changing this address.

An entry of '-EXT' after 'TEF20HI' entry(s) in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP56001 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP56001, 74.1 nsec
Data Format, Wordlength:	fixed-point, 24-bit
1 channel Hard Disk Record:	48 kHz 1024 pt FFT Accel. Disk File Rate: 46.8 Hertz/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 24 P, X, Y
Maximum Ext. Mem. Config:	16k x 24 P, X, Y
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	no
1 channel Hard Disk Playback:	48 kHz
Max. 2-chn. Spectrum Analyzer Rate:	48 kHz
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	90 dB
THD:	UA
Voltage Range:	±3V (line-level) ±1.41 (microphone)
Input Impedance:	UA
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	no
External Sample Clock:	no
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	48 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	90 dB
THD:	UA
Voltage Range:	±3V
Output Impedance:	UA
Reconstruction Filters:	yes, FC24, Cheby8
Sin(x)/x Filters:	yes
Programmable Sampling Rate:	no
External Sample Clock:	no
Max. Sampling Rate:	48 kHz
Min. Sampling Rate:	48 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The Techron TEF20HI system is shipped standard with 16k x 24 of program, X: data, and Y: data memory. Hypersignal uses less than 8k x 24 of program memory, and the full 32k x 24 of data memory for continuous analog I/O data buffers and file-based acceleration processing areas. Although the 32k x 24 of data memory cannot be expanded, and is thus limited by compared to other DSP56001-based systems, stimulus & response and accelerated convolution operation are fully functional. A maximum stimulus size of 16384 samples. Maximum stimulus size using the MLS function included with Hypersignal-Acoustic is 16383.

Additional Notes About Hypersignal Software Operation

1) Currently, external triggering operation is inactive with the Techron TEF20HI system in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

2) The Techron TEF20HI system supports programmable input source (line-level or microphone) and programmable gain selections. In the Analog Conversion function, enter channel list data as 'nS', where "S" is the data source, and can be 'm' for microphone input or 'L' for line-level input. For example, to acquire a stereo microphone waveform, you would enter '0m,1m' in the CHANNEL LIST field. Please note that input must either be completely microphone or completely line-level. For example, a CHANNEL LIST entry of '0m,1L' would be the same as '0m,1m'; an entry of '0L,1m' would be equivalent to '0L,1L'. In the Digital Oscilloscope, and Spectrum Analyzer functions, enter analog input as 'AnS', where "S" is the data source. For example, to display "mic a" input on the top trace, you would enter 'A0m' in the TRACE1 field. To display "line in b" input on the bottom trace, you would enter 'A1L' in the TRACE2 field.

To control the current gain settings (applicable for microphone input only), enter the desired values in the INPUT GAIN LIST field in the Analog Conversion function. This field accepts three values, which should be separated by commas: input channel 0 ("mic a"), input channel 1 ("mic b"), and also an output channel 0 ("line out") *software gain* value. The first two values should

be entered in dB, ranging from 0 to 60 dB. Since the TEF20HI circuitry supports 16 levels of programmable gain, your entries will be truncated to the nearest 4 dB increment. The third value should be entered as a *linear scale factor*, which can be used to control the line output level (default value is 1).

3) In the Digital Oscilloscope and Spectrum Analyzer instrument menus gain values can be entered as either full-scale voltage values, or as dB gain values. Prefixing a value with 'd' specifies dB; prefixing a value with 'v' or no prefix specifies full-scale voltage (default mode). For example, entry of

d20,d40

in the INPUT RANGE field would specify 20 dB of gain on trace 1 (physical channel specified in the TRACE1 field) and 40 dB of gain on trace 2 (physical channel specified in the TRACE2 field). Remember, when entering full-scale voltage ranges, you enter a smaller value to get increased gain. For example, since the TEF20HI microphone inputs have full-scale (zero-gain) inputs of about 0.15V, an entry of '.015' would specify a gain of 10.

4) When using the TEF20HI, there are 16 programmable gain values possible, ranging from 0 dB to 60 dB in 4 dB increments. The software quantizes entry to the nearest of these values

5) You should have legitimate signals connected between the microphone V+ and V- inputs to see correct gain operation. Measurements on unconnected microphone channels (ambient noise) or between single-ended inputs (V+ and GND) may not show correct gain values.

6) Output gain in the Analog Conversion menu can only be one value, is a linear software scaling factor (not a true hardware gain value) and applies only to the single output D/A channel on the TEF20HI.

3.20 Texas Instruments

Both the SWDS TMS320C25 and EVM TMS320C30 development boards, manufactured by Texas Instruments, are supported by Hypersignal. Both boards have distinguishing characteristics: the SWDS board was designed to support real-time emulation and breakpoints, requiring some unique software in order to achieve reliable operation, and the EVM design was purposefully limited in terms of memory and analog I/O in order to keep its cost very low. However, in both cases, the Hypersignal software works remarkably well with these boards, providing functionality that only marginally degrades in a few areas.

3.20.1 Texas Instruments SWDS Board (TMS320C25)

Basic Descriptive and Installation Details

Use a hardware designator of 'SWDS' in the Hypersignal software System Configuration menu. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled.

The SWDS boad contains the TMS320C25 16-bit fixed-point DSP. Its analog I/O capability is totally dependent—in terms of channels, resolution, and anti-alias filtering—on what is connected to the external ribbon cable connectors on the SWDS board. The SWDS board is a full-length board with 8-bit ISA bus XT-style connector.

Basic Hypersignal operation with the SWDS board is pre-configured to use the parallel A/D and parallel D/A on an AIB-2 board. To configure Hyperignal software for other resources or multichannel operation on the AIB-2 board, contact Hyperception or Signalogic about purchasing the DSP Source Code Interface for the TMS320C25 family (see section 4.1, Texas Instruments TMS320C25 DSP Source Code Interface, for more details). In the case of the AIB-2 board, one 16-bit A/D and D/A channel is available with hard disk transfer rates up to 58 kHz. The AIB-2 board has two resistor-programmable 8th-order Butterworth filters, one for anti-aliasing on the A/D input, and the other for D/A output smoothing. The AIB-2 board also contains a TMS29C18 codec and TMS32040 analog interface circuit.

Without any external AIB board connected, the SWDS may still be used as a "DSP Accelerator" for non-real-time file-based acceleration by the Hypersignal software.

Initial default Hypersignal memory base address for the Texas Instruments SWDS board is D000H. SWDS board documentation recommends a memory base address of D000H for IBM-AT and compatible machines, and an address of E000H for IBM-PC and compatible machines. See Users Manual section 2.5.2, Initializing the System Configuration Menu, under the MEM BASE ADDRESS field, for details on changing this address.

NOTE: In many late-model systems, part or all of the D000H and E000H 64 kbyte memory segments are needed for a network card, "SuperVGA" video card, and/or EMS memory. We suggest using the E000H segment for EMS memory, placing the video card in a mode that doesn't require extra memory, setting the network card segment address to D000H, and also setting the SWDS board memory base segment address to D000H. This is possible since the SWDS doesn't actually address memory until segment D200H, allowing simultaneous use of EMS memory and a network card (barely).

Also, be sure that you "block out" the region from D200-DFFF when using an expanded memory manager program such as EMM386.EXE. See section 2.1, Compatibility Considerations, for general information about DSP boards with memory-mapped interfaces, such as the DSP96002 system board.

Since the SWDS board is located inside the PC host, two 40-pin ribbon cables are used to connect to an "adaptor" or "pod" which plugs directly into a dual-inline or pin grid array socket on the AIB board—thus the AIB board is located externally to the PC host. Unfortunately, signal degradation at the full 40 MHz speed over the two 40-pin ribbon cables can cause serious ringing and timing glitches on the external interface. There are two possible solutions to this problem:

1) *Reduce the SWDS board oscillator value to 32 MHz* when using the AIB-2 (or any other external hardware connected over the twin ribbon cables). You should definitely change the 40 MHz oscillator to 32 MHz if you want to use the AIB-2 with Hypersignal. Also, oscillator values less than 32 MHz may not work! Memory timing is very touchy on the SWDS board.

2) *Terminate critical signals on the "adapter" pod with 220 or 330 ohm terminating resistors*. If you have the time to get this working, then in the long run this solution will be best because the board might be used at the full 40 MHz.

There are several other hardware and software considerations when using the SWDS board with Hypersignal that you should be aware of:

1) *You must bend up or clip pin 2 on U25 for Hypersignal to work!* Because operation has proven to be more reliable with this fix, we require that you do this in order to use the SWDS with Hypersignal software.

2) On the AIB-2 board you should check the following jumpers before operating (incorrect jumper settings could cause software to freeze up):

- E1 none (no jumper installed)
- E2 B installed (A, C not installed)
- E3 B-C connected (B connected to C)
- E4 B installed (middle peg connected to B)

On early-model AIB-2 boards jumper E3 has only three positions instead of four. To get correct Hypersignal operation with these boards, E3 should have B connected to the middle peg.

3) U34 PAL on the SWDS board should have a checksum of 35C1. This represents an official Texas Instruments PAL change that improved a marginal memory-access timing condition on the SWDS board—without this change, data brought up to Hypersignal might appear to have "spikes" on it.

4) Some changes have been made in Hypersignal TMS320C25 source code especially to accommodate the marginal memory-access timing on the SWDS board. These include an ADDK 0 instruction between indirect post-decrement or post-increment memory access and CALL instructions and performing two TBLR instructions in succession to pre-charge memory address lines. For other boards, you may want to remove the ADDK 0s if you are using the DSP Source Code Interface (see section 4.1, Texas Instruments TMS320C25 DSP Source Code Interface, below). The ADDK 0 instruction is used because it is a "true" NOP, where as the NOP instruction itself is actually a MAR * instruction in disguise, which still affects memory address signals.

5) The SWDS may not operate reliably at I/O bus speeds greater than 8 MHz. In most cases, if your host machine can operate at non-standard speeds (the original IBM I/O AT bus speed was 8 MHz), then it should also have system switches on the motherboard allowing you to slow it down if necessary.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a

matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via TMS320C25 code.

Hardware Ratings and Software Throughput Specifications

NOTE: In the specifications below, values are given for both the standard 16-bit parallel A/D on the AIB-2, and also the TLC3204x AIC. When different , these values are separated by a "/".

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C25, 125 nsec
Data Format, Wordlength:	fixed-point, 16-bit
1 channel Hard Disk Record:	58 kHz/19.2 kHz
1 channel Hard Disk Playback:	58 kHz/19.2kHz
Max. 2-chn. Spectrum Analyzer Rate:	75 kHz/19.2 kHz
1024 pt FFT Accel. Disk File Rate:	9.5 frames/sec
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	12k x 16 D; 12k x 16 P
Maximum Ext. Mem. Config:	12k x 16 D; 12k x 16 P
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	no
External Parallel Interface:	yes
Host Data Transfer Method:	Memory Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	1,16-bits/1,14-bits TLC3204x AIC
Codec channels:	1, 13-bit TMS29C18
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	UA / 70 dB
THD:	UA
Voltage Range:	±5V, ±10V; J2 / ±1.5V to ±6V; P3
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Butter8, RP, JB/yes, Cheby7
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	58 kHz/19.2 kHz
Min. Sampling Rate:	76.3 Hz/1 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 16-bit/1,14-bit TLC3204x AIC
Codec channels:	1, 13-bit TMS29C18
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	UA / 70 dB
THD:	UA
Voltage Range:	±5V / ±6V
Output Impedance:	UA
Reconstruction Filters:	yes, Butter8, RP, JB/yes, Cheby7
Sin(x)/x Filters:	no
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	58 kHz/19.2 kHz
Min. Sampling Rate:	76.3 Hz/1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The SWDS board allows a total of 24k x 16 of onboard memory. Hypersignal configures this memory for 8k x 16 of program and 16k x 16 of data. This allows basically normal function operation except for convolution/correlation acceleration and stimulus & response operation, which are disabled.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the SWDS board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

3.20.2 Texas Instruments EVM Board (TMS320C30)

Basic Descriptive and Installation Details

Use a hardware designator of 'EVMC30' in the Hypersignal software System Configuration menu. The 80286 INSTRUCTION SET field in this menu should be enabled (set to 'Y') if you have a 80286, 80386, or 80486 machine.

The EVM board contains the Texas Instruments TMS320C30 32-bit floating-point DSP, and one analog I/O channel. It is a 1/2 length board with 16-bit ISA bus AT-style connector.

The EVM board contains a TLC32044 AIC connected to one of the two TMS320C30 serial channels, which can be programmed from Hypersignal for different sampling rates and one of two different input gains. The TLC32044 AIC (analog interface chip) provides one channel of 14-bit A/D and D/A. The TLC32044 device allows the sampling rate to be programmed, 3 levels of input gain to be selected, and contains anti-aliasing and reconstruction bandpass filters that track with the programmed sampling rate. The sampling rate and gain are controllable from Hypersignal menus. One should be aware that programming of the sampling rate is not exact, because the TLC3204x device uses two 6-bit integer counters as "divisors" into a master clock frequency, which allows only a discrete number of possibilities. Normally, input levels can be selected to be ±1.5V, ±3V, and ±6V. Also, the highpass portion of the bandpass filters can be selectively disabled (although this must be done through the Hypersignal configuration file as opposed to data-entry menus). For TLC32044 devices, a sin(x)/x D/A output filter is also incorporated.

NOTE: Before running Hypersignal with the EVM board, you must download a small host-interface program using the Texas Instruments EVM loader. To do this, locate the Texas Instruments "EVMLOAD.EXE" file and enter the following DOS command:

EVMLOAD EVM_INIT.OUT <car ret>

The "EVM_INIT.OUT" file is contained on Hypersignal distribution diskette #2. You must enter the ".OUT" extension in the command. Also, the "RESVCT.001" and "RESVCT.002" files must be present on the same subdirectory as "EVMLOAD.EXE."

If you plan to make any changes to the TMS320C3x source code running on the EVM, please consult section 4.5, Texas Instruments TMS320C3x DSP Source Code Interface, for notes on ensuring that the "EVM_INIT.OUT" file is kept up to date along with any other changes.

Hypersignal uses a default I/O base address of 240H for the EVM board. Note that you *cannot* change this address from the System Configuration menu, as is the case with other boards. To change the base address, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal. See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the EVM board I/O base address in the "HSMACRO.CNF" (or "HSACOUS.CNF") file.

By editing the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly (outside of Hypersignal, with a text-editor), TLC32044 parameters such as anti-alias and reconstruction filter enable/disable, synchronous I/O, AUX input, etc. can be programmed differently from the Hypersignal default. See the "CONFIG.DOC" file (on distribution diskette #2) for details both on the default values and on the location and format of the AIC control word in the configuration file. The rated maximum sampling rate for the TLC32044 is about 20 kHz, although some devices will operate adequately at rates up to 30 kHz.

In the Digital Oscilloscope and Spectrum Analyzer functions, an INPUT RANGE field setting of 3 (±3 Volts) will allow accurate measurement of input voltages, but you should attempt to keep input below ±2.5V to avoid any non-linearities and distortion. If you change the INPUT RANGE field to 1.5, you will increase the gain of the AIC chip to 2 (input at ±1.5V full-scale). The TLC32044 device on the EVM board has been hard-wired for single-ended analog input operation. Of the 4 manufacturers of boards with TLC3204x devices currently supported by Hypersignal, only the EVM is not set for differential-ended operation. In all other cases, you would use values of 6 and 3 (±6 Volts and ±3 Volts, respectively) in the instrumentation functions.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -8191 to 8191 for 14-bit A/D) and applied digitally, via TMS320C3x code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	TMS320C30, 60 nsec
Data Format, Wordlength:	floating-point, 32-bit
1 channel Hard Disk Record:	19.2 kHz
1 channel Hard Disk Playback:	19.2 kHz
Max. 2-chn. Spectrum Analyzer Rate:	19.2 kHz
1024 pt FFT Accel. Disk File Rate:	UA
Spectrum Analyzer Noise Floor:	UA

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	16k x 32
Maximum Ext. Mem. Config:	16k x 32
Ext. Sync Serial Interface:	yes (one port available for
Ext Async. Serial Interface:	either sync. or async.)
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x AIC
Codec channels:	none
Expandable:	no
Simultaneous S/H	no
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±1.5V to ±3V; P2
Input Impedance:	UA
Anti-Aliasing Filters:	yes, Cheby7, JB (AIC)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	1, 14-bit TLC3204x AIC
Codec channels:	none
Expandable:	no
Simultaneous S/H:	no
Typical Dynamic Range:	70 dB
THD:	UA
Voltage Range:	±6V
Output Impedance:	UA
Reconstruction Filters:	yes, Cheby7, JB (AIC)
Sin(x)/x Filters:	yes, AIC (TLC32044)
Programmable Sampling Rate:	yes
External Sample Clock:	no
Max. Sampling Rate:	19.2 kHz
Min. Sampling Rate:	1 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The EVM board allows for 16k x 32 of memory (either program or data). There is no possiblity of memory expansion. Hypersignal uses less than 8k x 32 of the memory for program storage. The remaining 8k x 32 data memory amount limits the size of stimuli in stimulus & response operations, limits the maximum file-based FFT acceleration and Spectrum Analyzer FFT size to 1024, and disables convolution/correlation acceleration. Consult the product catalog for other supported TMS320C3x boards that provide much greater memory (and also analog I/O) capacity.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the EVM board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

Note that you *cannot* change the EVM I/O base address from the System Configuration menu, as is the case with other boards. To change the I/O base address, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal (with an ASCII text-editor). See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the EVM board I/O base address in the "HSMACRO.CNF" (or "HSACOUS.CNF") file.

3.21 Turtle Beach Systems

Hypersignal drivers support the Turtle Beach Systems Tahiti, Monterrey, and previous MultiSound versions. All of these boards look basically the same to the Hypersignal driver in terms of their host PC digital interfaces. Although these boards contain both I/O mapped and memory-mapped interfaces to the host PC (ISA bus), Hypersignal drivers use only the I/O mapped interface.

The MultiSound is no longer marketed by Turtle Beach Systems. The Monterrey is a Tahiti board with MIDI wavetable synthesis daughterboard (Rio board) included. Tahiti boards can be upgraded to allow MIDI wavetable synthesis at any time by purchasing the Rio board. Both Monterrey and Tahiti include an auxillary external stereo input (and auxillary input header), internal CD-ROM input header, internal joystick header, and external MIDI connector.

3.21.1 Turtle Beach Systems Tahiti (DSP56001)

Basic Descriptive and Installation Details

Use a hardware designator of 'TBS56' in the Hypersignal software System Configuration menu for the Turtle Beach Systems (TBS) MultiSound or Monterrey, and 'TBS56-B' for the TBS Tahiti. It does not matter whether the 80286 INSTRUCTION SET field in this menu is enabled or disabled. Discussion below mentions only the Tahiti board, but applies equally to the Monterrey board.

The Tahiti board contains the Motorola DSP56001 24-bit fixed-point DSP, and two analog I/O channels. It is a 2/3 length board with 8-bit ISA bus XT-style connector.

Basic Hypersignal operation with the MultiSound, Monterrey, and Tahiti boards utilizes the two (2) 16-bit sigma-delta analog I/O channels on the Tahiti board. The input channels are implemented using a Crystal Semiconductor CS5336 device, and the output channels using a CS4328. These devices offer more than 90 dB of dynamic range, and Turtle Beach Systems has achieved a board layout that provides excellent signal conditioning and noise floor performance. Sampling rates are software-selectable from Hypersignal menus as 11.025, 22.05, and 44.1 kHz. Additionally, Hypersignal sampling rate conversion functions can be used to derive other sampling rates from these basic rates. If you have the Hypersignal DSP56001 Source Code Interface (see section 4.3, Motorola DSP5600x DSP Source Code Interface), it is straightforward to utilize the built-in real-time filtering capability to achieve real-time sampling rate conversion. Also, the sampling rate oscillator on the Tahiti board can be replaced with a different value to achieve desired real-time sampling rates with optimum performance (no extra signal processing overhead).

Initial default Hypersignal I/O base address for the Tahiti board is 290H. As mentioned above, the Tahiti memory base address is not used by Hypersignal (although it is used by Turtle Beach software programs, and must be correctly configured for these programs if used. See the the TBS Tahiti Users Guide for detailed information). To change the I/O base address, you must edit the "HSMACRO.CNF" (or "HSACOUS.CNF") configuration file directly, outside of Hypersignal (with an ASCII text-editor). See the "CONFIG.DOC" file (on distribution diskette #2) for the exact location of the Tahiti board I/O base address in these files. See Users Manual section 2.5.2, Initializing the System Configuration Menu, for general information on changing base I/O addresses.

An entry of '-EXT' after the entry(s) for this board in the System Configuration menu will disable any built-in sampling rate programming and menu-entry range checking. In this case, you would be expected to supply an external clock or non-standard crystal oscillators and ensure that a matching value is entered in the software SAMPLING FREQUENCY menu fields in order to maintain correct time units display and usage. If you change oscillator values, then you will need to make minor modifica-

tions to the DSP5600x source code in order to program the sampling rate latch correctly (because the built-in Hypersignal programming is disabled by the '-EXT' selection). See section 4.3, Motorola DSP5600x DSP Source Code Interface.

Input and output offset and trigger value entries in the Analog Conversion function are active (enter in range -32767 to 32767 for 16-bit A/D) and applied digitally, via DSP56001 code.

Hardware Ratings and Software Throughput Specifications

PERFORMANCE SPECIFICATIONS

Digital Signal Processor:	DSP56001, 50 nsec
Data Format, Wordlength:	fixed-point, 24-bit
2 channel Hard Disk Record:	44.1 kHz
2 channel Hard Disk Playback:	44.1 kHz
Max. 2-chn. Spectrum Analyzer Rate:	44.1 kHz
1024 pt FFT Accel. Disk File Rate:	15 frames/sec
Spectrum Analyzer Noise Floor:	92 dB

DIGITAL SPECIFICATIONS

Standard Ext. Mem. Config:	8k x 24 P; 12k x 24 X,Y
Maximum Ext. Mem. Config:	8k x 24 P; 12k x 24 X,Y
Ext. Sync Serial Interface:	no
Ext Async. Serial Interface:	yes (joystick connector)
External Parallel Interface:	no
Host Data Transfer Method:	Prog. I/O Mapped

A/D SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H	yes
Typical Dynamic Range:	92 dB
THD:	.02% (unweighted)
Voltage Range:	±.12V, ±3.4V; P255
Input Impedance:	10k - 60k (35k default), RP
Anti-Aliasing Filters:	yes, SD
Programmable Sampling Rate:	yes, P3
External Sample Clock:	no
Max. Sampling Rate:	44.1 kHz
Min. Sampling Rate:	11.025 kHz
External TTL-Level Trigger:	no
Programmable Level Trigger:	no

D/A SPECIFICATIONS

Number of linear PCM channels:	2, 16-bits
Codec channels:	none
Expandable:	no
Simultaneous S/H:	yes
Typical Dynamic Range:	92 dB
THD:	.02% (unweighted)
Voltage Range:	±1.8V
Output Impedance:	UA
Reconstruction Filters:	yes, DS
Sin(x)/x Filters:	yes, DS
Programmable Sampling Rate:	yes, P3
External Sample Clock:	no
Max. Sampling Rate:	44.1 kHz
Min. Sampling Rate:	11.025 kHz
External TTL-Level Trigger:	no

Onboard Memory Usage

The Tahiti board has a total of 32k x 24 of onboard SRAM memory, in addition to the 512 x 24 of onchip SRAM built-in to the DSP56001. Hypersignal configures this external memory as 8k x 24 of P space (program) and 12k x 24 each of X and Y space (data). This is the maximum amount of memory available on the Tahiti board, and cannot be expanded. The configuration used by Hypersignal is the best tradeoff that can be obtained and still provided as much standard Hypersignal functionality as possible. However, the 24k x 24 total amount of data memory limits the use of the Tahiti as a DSP function accelerator (restricting maximum convolution and FFT sizes that can be used with acceleration), limits stimulus & response operation to a 4095 pt. stimulus, and limits spectrum analyzer operation to 1024 pts. dual-channel (512 pts. if transfer function measurements are en-

abled). It is important to note that Tahiti onboard SRAM runs at three (3) wait-states, which is a significant slow-down for the 40 MHz DSP56001 device. Hypersignal software seeks to minimize the impact of external SRAM wait-states by loading time-sensitive code sections into DSP56001 onchip SRAM whenever size and function permit.

Additional Notes About Hypersignal Software Operation

Currently, external triggering operation is inactive with the Tahiti board in the Analog Conversion function (i.e. entry of 'EXT' for TRIGGER field is ignored).

Input gain and range fields to do not cause the "digital input potentiometer" circuitry on the Tahiti board to be set. These fields should be set to the default voltage ranges listed in the specifications above in order to obtain accurate readouts in instrument or waveform displays.

For the Tahiti board, Hypersignal downloads a default DSP56TBS.OUT file, instead of the standard DSP56001.OUT file used for all other supported DSP5600x boards. This is done to accommodate the unusual memory map of the Tahiti board, which locates data memory at a high address, and not contiguously with the 512 x 24 word of internal DSP56001 SRAM.

this page intentionally left blank

4.0 DSP SOURCE CODE INTERFACE

Below please find information concerning the DSP Source Code Interface to Hypersignal software for various DSP chips. The disk containing the source code and support utilities for a given DSP chip family is not normally included in Hypersignal software shipments, unless purchased additionally or separately.

IMPORTANT NOTES: A signed Non-Disclosure Agreement (NDA) is required prior to shipping the DSP Source Code Interface for a given DSP chip. Bascially, the NDA requires work with the DSP Source Code to be for in-house R&D and product-development purposes, and prohibits using the DSP Source Code within, or to help develop, a product competing with Hypersignal. If you would like to see a copy of the NDA, please call and ask for one to be faxed to you. Also, the license agreement included with DSP Source Interfaces is shown below in section 4.0.1, Source Code License Agreement.

In order to make effective use of this flexible product, appropriate assemblers/linkers are required. C compilers are optional, but recommended if you plan to modify/add a substantial amount of code. However, compiler output—even that of the best optimizing compilers—is always at least 3-5x slower than hand-optimized code, which can make the difference between a system operating in real-time or not. Taking compiled code and optimizing only time-critical sections can prove an effective compromise. Assemblers, linkers, software simulators and C compilers for specific DSP types can be specified by part number; see the Hyperception and Signalogic price sheets.

In all cases, we have implemented and tested the DSP source code with the ***manufacturer's development tools*** (i.e. assembler, linker). The version of tools used for testing is indicated in the indvidual DSP sections below. Development tools from other vendors may be compatible, but we cannot guarantee this.

The DSP Source Code Interface includes both library routines and also real-time, interrupt-driven structures that allow foreground and background processing in conjunction with sample interrupts. Initially, the source code supports standard Hypersignal operation, including instrumentation displays, continuous transfer to/from disk, and file-based acceleration, with no programming required. In other words, if you were to assemble and link the source code for a specific DSP chip as it was shipped, without any changes, you would end up with the distribution object file that is sent with every basic Hypersignal package. However, the source code could then be modified and/or added to in order to enhance the baseline Hypersignal operations or add new functions entirely. Changes can be as simple as interfacing to new analog converters or as complicated as simulating and testing complete algorithms by transferring data between the DSP board and Hypersignal software resources (including hard disk and video display) using pre-defined buffer-switching mechanisms and real-time interrupt structures.

A simple example might be to create user-written DSP source code that might perform some algorithm or feature extraction. Following this, Hypersignal can be instructed to use the new executable DSP file(s) as needed, causing the new user-written algorithm(s) to execute. This type of modification process basically allows you to concentrate on the problem at hand, since the Hypersignal high-level host software performs all of the housekeeping associated with controlling the DSP board, downloading executable code, and transferring blocks of real-data between host and DSP board memory. Furthermore, if you maintain compatibility with internal Hypersignal data and waveform file formats, then more housekeeping tasks such as file creation, graphical analysis, etc. can be left to Hypersignal.

Following are some more detailed examples of using the DSP Source Code Interface. Lets suppose that you had one of the DSP boards that is supported by Hypersignal, but needed to interface the board to a target analog section of your own design. If you had the DSP source code for the appropriate chip family, this would be trivial. First, you would change the interrupt service routine for the Hypersignal function that you wanted to get working. Depending on the chip family there is generally a few different interrupt service routines for each function, organized by board manufacturer. The changes you would make here would be minor: ensuring that the code read your A/D port address instead of the supported board's standard address. Next, you might have to change the system initialization module if the interrupt source/level for your analog hardware is different. Everything else would be left alone: buffering of the data, transfer to the high-level host software, and processing of the data (although you might want to change that, too, at some point). Next, you would assemble the modules changed using the "ASM" batch file provided for each chip family. Finally, you would invoke the "LNK" batch file, and copy the resulting linker output file (".OUT",

".BIN", ".HEX", etc. depending on manufacturer convention) to your working Hypersignal directory. When you next invoked Hypersignal, this file would be ready for downloading and subsequent execution without any further effort.

As another example, suppose you wanted to take a supported board, and working with the board's standard analog I/O, implement all or part of a time domain algorithm on the board and stream the real-time results to hard disk for later evaluation and display. If you first make the assumption that when running this function from Hypersignal you will ask for two-channel operation, you could simply alter the foreground processing to apply your algorithm to every other sample in the input buffers as they become ready. This is possible because multichannel data in Hypersignal is always transferred and processed in interleaved form; indeed, the high-level host software can be instructed to acquire/generate N channel data whether the board has N channels or not. Whether this works depends totally on the low-level DSP source code. After modification of the foreground module, and/or addition of new, user-defined routines, subsequent execution of Hypersignal with 2-channel operation specified would produce raw input data on the first channel, and processed data on the second channel. FFT operation on such a waveform would be automatic, since Hypersignal still thinks this is a "stereo" waveform, allowing convenient comparison viewing and evaluation in the frequency domain as well.-1

As a third example, consider using a supported board to simulate an entire project, maintaining just enough of the Hypersignal interface and real-time data transfer mechanism to allow real-time display of data, debug modes, and possible streaming of data to disk. After transferring code to your target system (via EPROM), there can be no doubt as to whether encountered problems are due to the software or the target hardware design—the Hypersignal simulation can always be easily run at any time as a check against the software design.

Or, consider a case where the DSP Source Code Interface is used as the basis for software in a new product or system, with embedded intermediate data "probe" points and real-time transfer mechanisms left installed. By connecting a computer to the product in some way (RS-232 or parallel connection), a laptop or notebook computer can with Hypersignal installed could be brought into the field and used for testing, maintenance and/or data collection at any time.

Other examples include adding DSP source code to support multiprocessing, interconnections between multiple boards, and other peripheral bus standards such as DT-Connect, measuring algorithms or parts of algorithms against a particular real-time sampling rate, evaluating the effects of finite word-length, and more.

Working with the DSP Source Code Interface is very straightforward. The fact that we have written similar implementations for 7 different DSP chip families has forced us to standardize our library, from the function level down to the batch files used to facilitate the code modification process. For example, all DSP chip families have similar routines available for post-FFT processing: log, squareroot, arctan, etc. And, at a lower level, all "LNK.BAT" files produce a map file of the form "XXXLINK.MAP" where XXX is "T30" for Texas Instruments TMS320C3x, "M56" for Motorola DSP5600x, "N20" for NEC 77220, etc. Moreover, the modification and download process is also standardized between DSP families. For example, Hypersignal parses and downloads output files from each DSP manufacturer's linker to the DSP board for subsequent execution, obviating the need for user-written object file downloaders.

Most interrupt service routines fully save and restore machine context to protect anything they might modify, and most routines (such as the "snap-in" real-time filter routines that we have included in the source code library) also save and restore anything they might destroy. In all chip families the Digital Oscilloscope and Spectrum Analyzer functions provide the best examples of full context-save interrupt service routines combined with substantial foreground processing constructed from many of the routines in the source code library.

4.0.1 Source Code License Agreement

Compliance with the following license agreement is one of the provisions in the NDA agreement mentioned above, a signed copy of which is required before delivery of a DSP Source Code Interface can take place:

Hyperception, Inc. and Signalogic, Inc. reserve all rights to source code supplied with "Hypersignal" series software.

USE OF THIS SOURCE CODE BY YOU ("YOU" MEANS THE COMPANY AND/OR INDIVIDUALS THAT ARE USING THE "HYPERSIGNAL" SOFTWARE) FOR PURPOSES OF RESALE OR PROFIT, OR FOR INCLUSION IN, EITHER WHOLE OR IN PART, ANY OTHER SOFTWARE INTENDED FOR SALE OR PROFIT, IS PROHIBITED.

You are free to copy source code sections to make backup copies and to perform experimentation and testing during algorithm and product development and research. If it is discovered that you, through resale, through lending, or allowing through carelessness or inaction, have enabled another party to obtain possession of a section of the DSP source code marked with a

Hyperception or Signalogic copyright, which the other party is including within, or with, a product competitive with the "Hypersignal" series software, then HYPERCEPTION AND SIGNALOGIC ARE PREPARED TO PROSECUTE TO THE FULLEST EXTENT OF THE LAW.

YOU ACKNOWLEDGE THAT YOU HAVE READ THE ABOVE AGREEMENT, UNDERSTAND IT, AND AGREE TO BE BOUND BY ITS TERMS AND CONDITIONS. YOU FURTHER AGREE THAT IT AND THE NON-DISCLOSURE AGREEMENT (see above) IS THE COMPLETE AND EXCLUSIVE STATEMENT OF ANY AGREEMENT BETWEEN US, INCLUDING ANY OTHER ORAL OR WRITTEN COMMUNICATION.

4.1 Texas Instruments TMS320C25 DSP Source Code Interface

4.1.1 TMS320C25 Overview

The TMS320C25 Source Code Interface distribution diskette contains all TMS320C25 source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic TMS320C25 source code includes interrupt service routines for the LSI/Spectrum, Ariel DSP-16 and PC-C25 (formerly called the DSP-C25), Texas Instruments SWDS, and Burr-Brown PCI-20202C TMS320C25 boards.

As is, the TMS320C25 source code is written in modular form, and supports the basic "modes" of operation below:

'C25 Module	**Hypersignal Functions**	**oper. "mode"**
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the TMS320C25 boards and generally match up with the foreground processing modules above. Interrupt service routine module names will have "ISR" in the middle of the filename, after two letters indicating the hardware they support (ISRs are usually hardware-specific), and before a digit indicating the operating mode (see above) that they support, for example "LSISR0". Different interrupt service routines use different interrupt levels for different boards. Examine the interrupt vector and mask initialization in the "T25DEF" (non-executable definitions) and the "T25BEG" module, depending on board type (see host-shared variable "BOARD"), to locate the correct interrupt service routine for a particular function.

MODIFICATION OF EXISTING SOURCE CODE

If the user wishes to alter the performance of the code, he/she must keep in mind the necessary links between Hypersignal software and the TMS320C25 code. The two-way communication between the PC and TMS320C25 established in the Hypersignal software must not be changed, as the high-level host software expects to read/write certain memory locations. These memory locations are initialized by table reads from program memory inside module T25BEG, and are suffixed by 'ADD', as in MODADD, TRGADD, etc. *Do not insert any code of any type before these addresses that would cause them to move.* The TMS320C25 source code has been documented to aid the user in these areas.

FOREGROUND PROCESSING AND REAL-TIME INTERRUPT STRUCTURE

Currently, operating mode 5 (Spectrum Analyzer) is the best example of full implementation of foreground processing with complete context-save background interrupts. The Spectrum Analyzer source code represents a totally real-time system, with

samples buffered in a dual-buffer scheme in response to background “sample ready” interrupts, and buffers, when full, sent to FFT and post-FFT processing in the foreground. Use mode 5 source modules (“MODUL5" and modules it calls), and ”xxISR5" for one example of a real-time, interrupt-driven structure.

4.1.2 TMS320C25 Source and Support File List

Source code and related files included on the distribution diskette:

T25DEF.ASM	variable & constant definition
T25BEG.ASM	beginning vectors, including ISR vectors, and including initialization, including PC-written variables
LSISR0.ASM	A/D ISR for LSI/Spectrum board
ARISR0.ASM	A/D ISR for Ariel DSP-16 board
SWISR0.ASM	A/D ISR for Texas Instruments SWDS board
AIISR0.ASM	A/D ISR for Ariel PC-C25 (formerly DSP-C25) board (with TLC3204x AIC)
BBISR0.ASM B0ISR0.ASM B2ISR0.ASM	A/D ISRs for Burr-Brown PCI-20202C
LSISR1.ASM	D/A ISR for LSI/Spectrum board
ARISR1.ASM	D/A ISR for Ariel DSP-16 board
SWISR1.ASM	D/A ISR for Texas Instruments SWDS board
AIISR1.ASM	D/A ISR for Ariel PC-C25 (formerly DSP-C25) board (with TLC3204x AIC)
BBISR1.ASM	D/A ISR for Burr-Brown PCI-20202C
ISR2.ASM	Digital Scope ISR for all TMS320C25 boards.
ISR5.ASM	Spectrum Analyzer ISR for all TMS320C25 boards.
ISR6.ASM	Stimulus & Response ISR for all TMS320C25 boards.
MODUL0.ASM	foreground A/D processing (mode=0)
BBMOD0.ASM	foreground A/D processing (mode=0; Burr-Brown board only)
MODUL1.ASM	foreground D/A processing (mode=1)
MODUL2.ASM	foreground Digital Oscilloscope processing (mode=2)
MODUL3.ASM	FFT file-based acceleration processing (mode=3)
MODUL4.ASM	iFFT file-based acceleration processing (mode=4)
MODUL5.ASM	foreground Spectrum Analyzer process (mode=5)
MODUL6.ASM	foreground stimulus & response (mode=6)
MODUL7.ASM	convolution/correlation file-based acceleration processing (mode=7)
ARCTAN.ASM	arctangent routine
CONVLV.ASM	convolution/correlation routine
DIVIDE.ASM	fractional and integer divide routines
EXP2.ASM	exponentiation (base-2) routine (needed for fixed-point transfer function calculation)
FFT.ASM	variable N ($N=2^m$) FFT routine
FFTOUT.ASM	post-FFT processing (mag, phase, scaling)

FILTER.ASM	invokes correct filter for current channel
FIR.ASM	FIR filter routine
IIR2.ASM	IIR filter cascade struc 2 (modified direct-form)
IIR3.ASM	IIR filter cascade struc 3 (balanced pole-zero)
LOG2.ASM	log2 approximation routine
MAG.ASM	magnitude routine
MAX.ASM	find maximum routine
MIN.ASM	find minimum routine
MOVE.ASM	move data
PHASE.ASM	4-quadrant phase routine
SINTAB.ASM	sine value lookup-table for FFT use
SPAOUT.ASM	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.ASM	squareroot routine
WINDOW.ASM	apply pre-defined window (used with FFTs)
XFER.ASM	complex-ratio routine used for transfer function
ASM.BAT	batch file used to assemble any source module: type “ASM filename” (no extension needed)
LNK.BAT	batch file used to link all source modules: type “LNK”
T25LINK.CMD	link control file (batch file for linker)

The distribution disk does not initially contain all .OBJ files necessary to produce a complete baseline, default executable code module when the T25LINK.CMD control file (above) is invoked with the Texas Instruments “dsplnk” utility. Each .ASM module above must be assembled first.

4.1.3 TMS320C25 Modification Process

NOTE: Be sure to save a backup copy of TMS32025.OUT before running the “LNK” batch file in case your new code doesn’t work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to TMS320C25 source code, keeping in mind the necessary shared variables between Hypersignal and the TMS320C25. The file T25DEF.ASM is well-documented in this area.

2) Assemble all modules changed and any new modules created (using Texas Instruments “COFF” assembler) to ensure that a new .OBJ (object) file is generated for each altered module. For convenience, an “ASM.BAT” batch file has been provided that can be invoked as:

ASM filename <car ret>

to assemble source code modules (entering the “.ASM” extension after “filename” is optional). This file executes a

dspa -l -v25 %1

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the T25LINK.CMD control file (use an ASCII text-editor to update the T25LINK.CMD file).

3) Re-link all modules using Texas Instruments TMS320C25 "dsplnk" linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .OBJ files. LNK.BAT executes a

dsplnk T25LINK.CMD

statement, where "T25LINK.CMD" is an ASCII linker-command file that specifies all .OBJ files to be linked by the dsplnk.exe program, as well as specific information about memory addresses for some sections. Note that all executable code that is re-locatable starts at 400H. All code has been tested using version 5.05 of "dspa.exe" and "dsplnk.exe".

4) Copy the resulting "TMS32025.OUT" file to directory with Hypersignal software, run Hypersignal, and invoke the appropriate functions. Hypersignal reads the "COFF" file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING TMS320C25 SOURCE CODE

1) If you need to change default buffer addresses, please consult the comments and suggestions in the T25DEF.ASM file.

2) The LNK.BAT file produces a "T25LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

3) The T25LINK.CMD file creates an uninitialized "hole" from 20H through 3FFH that can be used for monitor and debugger programs. Because this section is uninitialized, the Hypersignal COFF loader will not write anything to this area.

4) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". The value is in Hz (Hertz); note that for TMS320C25 boards, which use 16-bit wide memory accesses, the maximum value that can be represented is limited to 65535. This value is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED TMS320C25 SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as "new.out", after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For TMS320C25 boards, you should avoid writing to program memory between 20H and 3FFH, which is typically left open for TMS320C25 debuggers, and 42DH and 4FFH, which is used by Hypersignal to store host-DSP communication variables and structures. Also, note that if Hypersignal DSP TMS320C25 code is allowed to run, it stores local variables in onchip data memory locations 0-127, location 2FF, and 896 through 1023 (filter coefficients and delay elements).

4.2 AT&T WE DSP32 DSP Source Code Interface

4.2.1 WE DSP32 Overview

The DSP32 Source Code Interface distribution diskette contains all AT&T WE DSP32 source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic DSP32 source code includes hardware support code for Communications Automation and Control (CAC) and Burr-Brown ZPB32 WE DSP32 boards.

As is, the DSP32 source code is written in modular form, and supports the basic “modes” of operation below:

DSP32 Module	Hypersignal Functions	oper. “mode”
DSP32.S	Analog Conversion	mode=2
DSP32.S	Analog Conversion, Playback from Waveform Display	mode=1
DSP32.S	Digital Oscilloscope	mode=3
DSP32.S	FFT file-based acceleration	mode=5
DSP32.S	iFFT file-based acceleration	mode=6
DSP32.S	Spectrum Analyzer	mode=4
DSP32.S	User-Defined function	mode=n

4.2.2 WE DSP32 Source and Support File List

Source code and related files included on the distribution diskette:

DSP32.S	contains sections for all functions
GETSIM.COM	generates new DSP32.BIN file from DSP3SIM simulator ASCII output file
GETPRG.COM	generates new DSP32.BIN file from DSP32 board program memory (assumes board is DS3EMU emulator-compatible)

4.2.3 WE DSP32 Modification Process

NOTE: Be sure to save a backup copy of DSP32.BIN before running the “LNK” batch file in case your new code doesn’t work.

If you wish to alter the performance or function of the code, you must keep in mind the necessary links between the high-level Hypersignal host software and WE DSP32 code. The two-way communication between the PC and DSP32 established in the high-level host Hypersignal software must not be changed, as the high-level software expects to read/write certain memory locations. The WE DSP32 source code has been documented to aid the user in this area.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to DSP32 source code, keeping in mind the necessary links between Hypersignal and the DSP32. All steps below refer to source code as filename.s. If you start with the source code included on the distribution disk, then filename.s = DSP32.S in the steps below.

The user may find it convenient to create a batch file to execute the following steps 2-6.

2) Re-assemble the source code using the AT&T "make utility" (assembler and linker) by typing:

DSP3MAKE -m2 filename.s <car ret>

Add the switch "-l" after "-m2" if you desire a listing file output. Add the switch "-s" if simulator testing will be required—but you will need to move the current FFT window data space at on-chip RAM F000H (memory mode 2) because the simulator wants to store breakpoint code at FE00H. The "asm.bat" file does the same thing as the DSP3MAKE command listed above; type:

ASM filename <car ret>

3) Run the AT&T simulator program to create a file of op-codes. Type:

DSP3SIM filename.o <car ret>

where filename is the name of the source program.

After the "$im:" prompt, type

mem[0:last_byte].b>"DSP32.BYT" <car ret>

where "last_byte" is the address of the last byte of program code, or enter a value, for example 0x1000, that ensures all program code is written to disk (The default amount for the source code in its initial form is 0x1000. If you add a significant amount of code, or change the "buf0adr" base address value, you may need to change the amount of simulator data written to the "dsp32.byt" file.)

To exit the simulator, type:

quit <car ret>

4) Run the GETSIM program; type:

GETSIM <car ret>

GETSIM expects a file with the name of "dsp32.byt" although this may be changed by altering the source for GETSIM, getsim.pas. Of course, step 3 above must use the new filename as well.

NOTE: Step 4 converts "DSP32.BYT" from an ASCII file to a binary file and saves it under the name "DSP32.BIN". Any method may be used to convert the simulator output to a binary file of consecutive program op-code bytes; the GETSIM program is just one way.

5) Copy the DSP32.BIN file to the directory with Hypersignal software. Hypersignal DSP software will use this file when executing DSP32 functions.

NOTE: The GETPRG program reads program code directly from DSP32 board memory and stores it in the file "DSP32.BIN". The source for GETPRG, "GETPRG.PAS", is included on the distribution disk and may be modified to change the number of bytes written out, etc. This represents an alternative method to the use of the simulator as described above; for example, the DSP3EMU emulator is used to load the program (created by the DSP3MAKE utility) into DSP32 board memory, then the GETPRG program is run after exiting the emulator, generating a DSP32.BIN file for use by Hypersignal.

MISCELLANEOUS NOTES ABOUT CHANGING DSP32 SOURCE CODE

1) When you add code to the DSP32.S file, be aware of the "0x1000" boundary. Hypersignal-Macro high-level code assumes that the start of various data buffers is 0x1000, so code that lands beyond this value could be overwritten. You can check the "DSP32.L" list file The "buf0_adr" variable (location 0x1A) can be changed to instruct the high-level software to make more room for program code. However, a better approach, if possible, might be to temporarily de-activate some functions to make room for additional code, so that the amount of data space beyond 0x1000 can be preserved.

2) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field). Note that a maximum value of 65536 can be represented, because integer values on the DSP32 are 16-bits.

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

4.3 Motorola DSP5600x DSP Source Code Interface

4.3.1 DSP5600x Overview

The DSP5600x Source Code Interface distribution diskette contains all Motorola DSP5600x source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic DSP5600x source code includes interrupt service routines for the LSI/Spectrum, Motorola ADS (with external ADC56000 or 56ADC16 analog I/O options), Ariel PC-56 and DSP-56 boards, and the Techron TEF-20 system.

As is, the DSP5600x source code is written in modular form, and supports the basic "modes" of operation below:

'5600x Module	Hypersignal Function	oper. "mode"
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the DSP5600x board and generally match up with the foreground processing modules above. Interrupt service routine module names consist of “ISRn”, where n indicates the operating mode (see above) that the routine supports, for example “ISR0" for operating mode zero. Inside each interrupt service routine there are different entry points that contain hardware-specific input/output code for individual boards. These hardware-specific sections are kept as small as possible, so that the great majority of the routine is common for all boards. Examine the interrupt vector and mask initialization in the ”M56DEF" (non-executable definitions) and “M56BEG” modules, depending on board type (see host-shared variable “BOARD”), to locate the correct interrupt service routine for a particular function.

MODIFICATION OF EXISTING SOURCE CODE

If the user wishes to alter or add code functionality, or improve the functionality and/or performance of existing code, he/she must keep in mind the necessary intercommunication variables between Hypersignal software and DSP5600x code. The two-way communication between the host-PC and DSP5600x set up in the Hypersignal software must not be changed, as the software expects to read/write certain memory locations. These memory locations reside at locations “intercom” and “intermod” and are clearly commented inside M56DEF. *Do not insert any code of any type that would change the current addresses of these variables.* If necessary, you may add further communication variables for your own high-level code. *You may change* the values of ANABFK, WNDBFK, RIBUFK, and MPBUFK (time data buffer, FFT window data buffer, real/imaginary data buffers, & magnitude/phase data buffers, respectively) if it becomes necessary to alter memory allocation in your DSP5600x code and communicate these changes to the host PC). The DSP5600x source code has been well-documented to aid the user in these areas.

FOREGROUND PROCESSING AND REAL-TIME INTERRUPT STRUCTURE

Currently, operating mode 5 (Spectrum Analyzer) is the best example of full implementation of foreground processing with complete context-save background interrupts. The Spectrum Analyzer source code represents a totally real-time system, with samples buffered in a dual-buffer scheme in response to background “sample ready” interrupts, and buffers, when full, sent to FFT and post-FFT processing in the foreground.

Use mode 5 source modules (“MODUL5" and modules it calls), and ”ISR5", for a good example of a real-time, interrupt-driven structure.

4.3.2 DSP5600x Source and Support File List

Source code and related files included on the distribution diskette:

M56DEF.ASM	variable & constant definition module
M56VEC.ASM	reset and interrupt vector locations, and host interface
M56BEG.ASM	initialization, including variables, operating modes, interrupt vectors, and analog I/O drivers
ISR0.ASM	A/D ISR for all boards
ISR1.ASM	D/A ISR for all boards
ISR2.ASM	Digital Oscilloscope ISR for all boards
ISR5.ASM	Spectrum Analyzer ISR for all boards
ISR6.ASM	Stimulus & Response ISR for all boards
MODUL0.ASM	foreground A/D processing (mode=0)
MODUL1.ASM	foreground D/A processing (mode=1)
MODUL2.ASM	foreground Digital Oscilloscope processing (mode=2)
MODUL3.ASM	FFT file-based acceleration processing (mode=3)

4.3.2 DSP5600x Source and Support File List

MODUL4.ASM	iFFT file-based acceleration processing (mode=4)
MODUL5.ASM	foreground Spectrum Analyzer processing (mode=5)
MODUL6.ASM	foreground stimulus & response processing (mode=6)
MODUL7.ASM	convolution/correlation file-based acceleration processing (mode=7)
ARCTAN.ASM	arctangent routine
CONVLV.ASM	convolution/correlation routine
DIVIDE.ASM	fractional and integer divide routines
EXP2.ASM	exponentiation (base-2) routine (needed for fixed-point transfer function calculation)
FFT.ASM	FFT/inverse FFT routine
FILTER.ASM	filter control routine for real-time filtering
FIR.ASM	FIR filter routine
IIR2.ASM	IIR (cascade sections, modified direct form) filter routine
IIR3.ASM	IIR (cascade sections, balanced pole-zero form) filter routine
LOG2.ASM	log2 approximation routine
MAG.ASM	magnitude routine (complex input)
PHASE.ASM	4-quadrant phase routine (complex input)
FFTOUT.ASM	post-FFT processing (mag, phase, scaling)
SPAOUT.ASM	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.ASM	squareroot routine
WINDOW.ASM	window routine: applies PC-downloaded window for Spectrum Analyzer and FFT
XFER.ASM	complex-ratio routine used for transfer function
XMAX.ASM	location of maximum value in X:buffer data
XMIN.ASM	location of minimum value in X:buffer data
YMAX.ASM	location of maximum value in Y:buffer data
XXMOVE.ASM YYMOVE.ASM XYMOVE.ASM YXMOVE.ASM LMOVE.ASM	move routines between X: and Y: memory
M56LINK.CTL	master linker control file
M56MEM.CTL	another linker control file, containing memory and I/O access wait-state value for all boards execpt TBS
M56TBS.CTL	another linker control file, containing memory and I/O access wait-state value for all TBS boards
ASM.BAT	batch file that can be used to assemble individual source code modules (use in form “ASM filename”)
LNK.BAT	batch file that performs linking (use in form “LNK”)

The distribution disk does not initially contain all .CLN files necessary to produce a complete baseline, default executable code module when the M56LINK.CTL control file (above) is invoked with the Motorola DSPLNK utility. Each .ASM module above must be assembled first.

4.3.3 DSP5600x Modification Process

NOTE: Be sure to save a backup copy of DSP56001.OUT (and DSP56TBS.OUT, if you are using a Turtle Beach Systems board) before running the "LNK" batch file in case your new code doesn't work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to DSP5600x source code, keeping in mind the necessary links between Hypersignal and the DSP5600x (see discussion above, and M56DEF.ASM module, which is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .LNK (object) file is generated for each altered module. For convenience, an "ASM.BAT" batch file has been provided that you may use in the form of:

ASM filename <car ret>

to assemble source code modules (entering an ".ASM" extension after "filename" is optional). This file executes a

asm56000 -B -L %1

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the M56LINK.CTL control file (use an ASCII text-editor to update the M56LINK.CTL file).

3) Re-link all DSP5600x modules using the Motorola DSPLNK linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .LNK files. LNK.BAT executes a

dsplnk -Z -RM56MEM.CTL -BDSP56001.OUT -MM56LINK.MAP -FM56LINK.CTL

statement, where "M56LINK.CTL" is an ASCII linker-control file that specifies all .LNK files to be linked by the DSPLNK.EXE program. Note that all executable code that is re-locatable starts at after the M56VEC and M56DEF modules, which contain some absolutely located data (which is concerned with host communications). All code has been tested using version 4.1 of Motorola's ASM56000 and DSPLNK.

4) Copy the resulting "DSP56001.OUT" (or DSP56TBS.OUT, if you are using a Turtle Beach Systems board) file to directory with Hypersignal, re-invoke the Hypersignal software, and run the appropriate function(s). Hypersignal reads the ".OUT" file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

5) If you prefer, change the lnk.bat file to produce some other "filename.out" file, instead of using the same name as the Hypersignal default. You can instruct Hypersignal to use non-default filenames in the System Config menu (see below for details).

MISCELLANEOUS NOTES ABOUT CHANGING DSP5600x SOURCE CODE

1) If you need to change default buffer addresses, please consult the comments and suggestions in the M56DEF.ASM file.

2) The LNK.BAT file produces a "M56LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

3) NOTE: The R7 and M7 registers are dedicated to the Hypersignal-Macro host-interface, and should not be used in user-defined code, for any reason, ever. This may cause a problem if you are using a C-compiler. If you need to free up the n7 register set, we suggest that you expand the fast interrupts used in the Hypersignal-Macro host-interface definition to long interrupts, and save registers used temporarily on the stack. This will result in a performance decrease, but may be necessary in order to take advantage of the development time saved by using a high-level language to program the DSP5600x.

4) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ fields).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED DSP5600x SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as "new.out", after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For DSP5600x boards, you should avoid using the program memory area between 20H and 0FFH, which is used by Hypersignal to implement a host-interface protocol (20H-3FH), store host-DSP communication variables and structures (40H-7FH) and local DSP variables (80H-8FH). Also, note that for some DSP5600x boards, Hypersignal performs a 'pseudo-reset', in which a host interface vector is used to reset the chip, peripherals, and interrupts, and then jump to the actual reset vector at location zero.

4.4 AT&T WE DSP32C DSP Source Code Interface

4.4.1 WE DSP32C Overview

The DSP32C Source Code Interface distribution diskette contains all AT&T WE DSP32C source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic DSP32C source code includes interrupt service routines for the Ariel DSP-32C and PC-32C (with various analog I/O options), AT&T, CAC (both AT-DSP32C and AC5 series), Burr-Brown (Intelligent Instrumentation) ZPB34, MTT Lory DSP4200, LSI/Spectrum, National Instruments AT-DSP2200, and Signalogic Sig32C series boards.

As is, the DSP32C source code is written in modular form, and supports the basic "modes" of operation below:

DSP32C Module	Hypersignal Function	oper. "mode"
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the DSP32C boards and generally match up with the foreground processing modules above. Interrupt service routine module names consist of "ISRn", where n indicates the operating mode (see above) that the routine supports, for example "ISR0" for operating mode zero. Inside each interrupt service routine there are different entry points that contain hardware-specific input/output code for individual boards. These hardware-specific sections are kept as small as possible, so that the great majority of the routine is common for all boards. Examine the interrupt vector and mask initialization (”itable" and PCW register) in RESET.S and W32BEG.S to locate the correct interrupt service routine for a particular function.

4.4.2 WE DSP32C Source and Support File List

Source code and related files included on the distribution diskette:

RESET.S	reset vector, wait-state, and "page register" initialization
W32DEF.S	variable & constant definition module
W32BEG.S	initialization, including ISR vectors
ISR0.S	A/D to disk ISR for all DSP32C boards
ISR1.S	D/A from disk ISR for all DSP32C boards
ISR2.S	Digital Oscillscope ISR for all DSP32C boards
ISR5.S	Spectrum Analyzer ISR for all DSP32C boards
ISR6.S	stimulus & response ISR for all DSP32C boards
ARI_TX.S	analog I/O output ISR for special-case boards only
MODUL0.S	foreground A/D to disk processing (mode=0)
MODUL1.S	foreground D/A from disk processing (mode=1)
MODUL2.S	foreground Digital Oscilloscope processing (mode=2)
MODUL3.S	FFT file-based acceleration processing (mode=3)
MODUL4.S	iFFT file-based acceleration processing (mode=4)
MODUL5.S	foreground Spectrum Analyzer process (mode=5)
MODUL6.S	foreground stim & response process (mode=6)
MODUL7.S	convolution/correlation file-based acceleration (mode=7)
ARCTAN.S	arctangent routine
CONVLV.S	convolution/correlation routine
DIV.S	divide routine
FFT.S	variable N (N=2^FFT_order) FFT routine

FFTOUT.S	post-FFT processing (mag, phase, etc.)
FILTER.S	filter control routine for real-time filtering
FIR.S	FIR filter routine
IIR2.S	IIR (cascade sections, modified direct form) filter routine
IIR3.S	IIR (cascade sections, balanced pole-zero structure) filter routine
LOG10.S	log (base 10) approximation routine
MAXFIND.S	find maximum-in-buffer routine
MINFIND.S	find minimum-in-buffer routine
MAG.S	magnitude routine (complex input)
PHASE.S	4-quadrant phase routine ($-\pi$ to $+\pi$; complex input)
SPAOUT.S	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.S	squareroot routine
WINDOW.S	apply pre-defined window (used with FFTs)
XFER.S	complex-ratio routine used for transfer function
ASM.BAT	batch file used to assemble any source module: type "ASM filename" (no extension needed)
LNK.BAT	batch file used to link all source modules: type "LNK"
W32LINK.CTL	link control file (command file for linker)

The distribution disk does not initially contain all .O files necessary to produce a complete baseline, default executable code module when the W32LINK.CTL control file (above) is invoked with the AT&T d3ld utility. Each .S module above must be assembled first.

4.4.3 WE DSP32C Modification Process

NOTE: Be sure to save a backup copy of DSP32C.OUT before running the "LNK" batch file in case your new code doesn't work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to DSP32C source code, keeping in mind the necessary shared variable links between Hypersignal and the DSP32C (see discussion above and W32DEF.S module, which is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .O (object) file is generated for each altered module. For convenience, an "ASM.BAT" batch file has been provided that you may use in the form of:

ASM filename <car ret>

to assemble source code modules (*do not enter a ".S" extension after "filename"*). This file executes a

d3as -Q -l %1.s

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the W32LINK.CTL control file (use an ASCII text-editor to update the W32LINK.CTL file).

3) Re-link all DSP32C modules using the AT&T DL3D linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .O files. LNK.BAT executes the following statements:

```
D3LD -a -m -o DSP32C.OUT W32LINK.CTL > W32LINK.MAP
@COPY W32LINK.MAP+EOF W32LINK.MAP
@D3NM -f DSP32C.OUT > TEMP
@COPY W32LINK.MAP+TEMP W32LINK.MAP
@DEL TEMP
```

where "W32LINK.CTL" is an ASCII linker-control file that specifies all .O files to be linked by the d3ld.exe program, and W32LINK.MAP is a map file showing variable and module addresses and lengths (W32LINK.MAP is created using DOS file-redirection commands, since the AT&T linker does not support a map file output option directly; see notes below). Note that W32LINK.CTL places the reset vector module at zero, and shared variables at 0x100, forcing all relocatable code to start after the variable declarations. All code has been tested using version 1.3.1 of d3as and d3ld.

4) Copy the "DSP32C.OUT" file to the directory with the Hypersignal HSxxxxx.OVR and HSxxxxx.EXE executable files, run Hypersignal, and invoke the appropriate functions. Hypersignal reads the ".OUT" file (which is in "COFF" format) directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING DSP32C SOURCE CODE

1) Hypersignal DSP32C code uses r19 to store a "page register value" (initially written by the high level software to DSP32C onchip RAM address 00FCH) so that Hypersignal DSP32C code can execute on different DSP32C boards that use different memory modes (which sometimes causes instructions to be in different 64K segments). *You should not change r19 at any time, for any reason.*

2) Hypersignal DSP32C code uses r18 as a system stack pointer for both foreground and interrupt processing code. When storing CAU registers on the "stack", use the following sequence:

```
*r18++ = rN1e
*r18++ = rN2e
       -
       -
       -
*r18++ = rNNe
```

When restoring CAU registers from the stack, use the following sequence:

```
rN1e = r18 - 4
rNNe = *rN1—
  -
  -
  -
rN2e = *rN1—
rN1e = *rN1
r18e = r18 - K        /* K = 4 x (number of registers stored) */
```

There are three remarkable things about this process. First, the stack is always expected to increment and decrement in 4-byte chunks (always save/restore extended registers). Second, the stack is always expected to point to the *next available (free) location*. Third, when restoring, you can't simply do an r18e = r18 - 4 instruction and begin popping rXX = *r18— instructions, because an interrupt might occur sometime after the r18 - 4 instruction and before you finish popping; the interrupt routine, if it also used r18 as a stack pointer (which all types of routines should be able to do of a legitimate stack pointer), might then clobber a stored register (or more than one) that hadn't been popped yet.

3) Since AT&T dedicates r20-r22 to special purposes, and Hypersignal DSP32C code uses r19 and r18 as system memory-page register and stack pointer, respectively, this means that *r18-r22 are reserved in the Hypersignal DSP32C source code and should not be used arbitrarily for any reason, ever.*

4) Hypersignal DSP32C source code is built with the "ASM" and "LNK" batch files, which invoke the AT&T "d3as" and "d3ld" assembler and linker utilities, respectively. By examining the ASM and LNK batch files, you should be able to construct an equivalent method of building the code using the AT&T "make" utility instead; however, producing a memory map file will be difficult. Obscure DOS commands have been used to create a map file in the LNK.BAT file, since the AT&T utilities will only print map output to the screen. The map file produced has a filename of "W32LINK.MAP"; you can use this file to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

5) Because the different DSP32C boards supported use different memory modes, Hypersignal DSP32C source code does not specify any memory mode of operation. In fact, the "LNK" batch file invocation of the AT&T linker (d3ld) does not specify any memory mode or configuration, causing the linker to assume contiguous memory starting at zero. This is possible because:

- The Hypersignal DSP32C source code always assumes at least 800H of read/write memory available at location zero (true for all memory modes). This area is used for the reset vector, storage of the "page register" value, storage of shared variables between host PC and DSP32C, storage of most DSP32C local variables, and system stack.
- Use of pc-relative "pccall" and "pcgoto" instructions that interpret procedure entry points and branch destinations as *offsets from the current pc (program counter) value,* rather than absolute constants.
- Use of r19 as a pre-defined "page register" for setting the upper 8 bits of registers being used to access local variables and interrupt table vectors not inside the 0-7FFH memory area. See discussion in 1), above.

6) The DSP32C apparently has a bug in the "call(rNN)" instruction in that only the least-significant 16 bits of the current program counter value are stored in rNN before the branch is taken. This is why all Hypersignal DSP32C code "pccall(rNN)" instructions perform a rNNe = rNN | r19 (logical "OR" with r19) as their latent instruction. We highly recommend that any calls you add to the source code do likewise.

7) In addition to setting them directly in the DSP32C source code, IOC register settings can be controlled from the high-level software in order to accommodate non-standard analog I/O connected to DSP32C serial ports. Inside the Hypersignal-Macro configuration file there is a 32-bit integer IOC value for every DSP32C board currently supported by Hypersignal-Macro—see the "CONFIG.DOC" file on disk #2 for a list of the sequence of variables in the configuration file. Depending on the board currently in use, one of these values is sent down to the DSP32C board whenever a Hypersignal-Macro real-time function requiring analog I/O is invoked. The corresponding DSP source code variable is "IOC_reg", which is normally used directly to set the IOC register (unless changed by the user in the DSP source code).

8) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED DSP32C SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as "new.out", after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For DSP32C boards, you should avoid using the program memory area between 100H and 2FFH, which is used by Hypersignal to store host-DSP interface variables and structures (100H-1FFH) and local DSP variables (200H-3FFH). Also, note that location 0FCH is used to store the page register value referred to above. This value is available for use; you can use it to ensure that your code attempts to execute in a memory area that physically exists on the DSP32C board.

4.5 Texas Instruments TMS320C3x DSP Source Code Interface

4.5.1 TMS320C3x Overview

The TMS320C3x Source Code Interface distribution diskette contains all Texas Instruments TMS320C3x source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic TMS320C3x source code includes interrupt service routines for the LSI/Spectrum system, ASPI Banshee, Sonitech Spirit-30, DSP Research TIGER 30, MTT Lory DSP4100, Domain Technologies DSPCard, Innovative Integration PC31, and Texas Instruments EVM boards.

As is, the TMS320C3x source code includes both library routines and also real-time interrupt driven “modes” of operation that use foreground processing and interrupt service routine processing to implement standard Hypersignal functions. These functions have been written in modular form, and support the basic “modes” of operation below:

‘C3x Module	Hypersignal Function	oper. “mode”
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the TMS320C3x boards and generally match up with the foreground processing modules above. Interrupt service routine module names consist of “ISRn”, where n indicates the operating mode (see above) that they support, for example “ISR0". Inside each interrupt service routine there are different entry points that contain hardware-specific input/output code for individual boards. These hardware-specific sections are kept as small as possible, so that the great majority of the routine is common for all boards. Examine the interrupt vector and mask ini-

tialization in the initial setup code in ”T30BEG", depending on board type (see host-shared variable “BOARD”), to locate the correct interrupt service routine for a particular function.

MODIFICATION OF EXISTING SOURCE CODE

If the user wishes to alter the performance of the code, he/she must keep in mind the necessary links between Hypersignal software and the TMS320C3x code. The two-way communication between the PC and TMS320C3x processor established in the Hypersignal software must not be changed, as the high-level software expects to read/write certain memory locations. These memory locations are initialized by the high-level software before any code executes on the DSP board (some locations are actually read by the high-level software as well, allowing you to change such “constants” as buffer base addresses). *Do not insert any code of any type before these addresses.* The TMS320C3x source code has been documented to aid the user in these areas.

FOREGROUND PROCESSING AND REAL-TIME INTERRUPT STRUCTURE

Currently, operating mode 5 (Spectrum Analyzer) is the best example of full implementation of foreground processing with complete context-save background interrupts. The Spectrum Analyzer source code represents a totally real-time system, with samples buffered in a dual-buffer scheme in response to background “sample ready” interrupts, and buffers, when full, sent to FFT and post-FFT processing in the foreground. The dual-buffer communication between the host and DSP is fully synchronized using “interlock” buffer flags. Use mode 5 source modules (“MODUL5" and routines it calls) and ”xxx_ISR5" for one example of a real-time, interrupt-driven structure.

4.5.2 TMS320C3x Source and Support File List

Source code and related files included on the distribution diskette:

T30DEF.ASM	variable & constant definition module
T30BEG.ASM	initialization, including ISR vectors
ISR0.ASM	A/D to disk ISR for all boards
ISR1.ASM	D/A from disk ISR for all boards
ISR2.ASM	Digital Scope ISR for all boards
ISR5.ASM	Spectrum Analyzer ISR for all boards
ISR6.ASM	stimulus & response ISR for all boards
MODUL0.ASM	foreground A/D to disk processing (mode=0)
MODUL1.ASM	foreground D/A from disk processing (mode=1)
MODUL2.ASM	foreground Digital Oscilloscope processing (mode=2)
MODUL3.ASM	FFT file-based acceleration processing (mode=3)
MODUL4.ASM	iFFT file-based acceleration processing (mode=4)
MODUL5.ASM	foreground Spectrum Analyzer process (mode=5)
MODUL6.ASM	foreground stimulus & response process (mode=6)
MODUL7.ASM	convolution/correlation file-based acceleration (mode=7)
ARCTAN.ASM	arctangent routine
CONVLV.ASM	convolution/correlation routine
DIVF.ASM	floating-point divide routine
FFT.ASM	variable N ($N=2^m$) FFT routine
FFTOUT.ASM	post-FFT processing (mag, phase, etc.)
LOG.ASM	log approximation routine

MAG.ASM	magnitude routine (complex input)
MAX.ASM	find maximum-in-buffer routine
MIN.ASM	find minimum-in-buffer routine
PHASE.ASM	4-quadrant phase routine ($-\pi$ to $+\pi$; complex input)
SINTAB.ASM	sine value lookup-table for FFT use
SPAOUT.ASM	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.ASM	squareroot routine
WINDOW.ASM	apply pre-defined window (used with FFTs)
FILTER.ASM	filter control routine for real-time filtering
FIR.ASM	FIR filter routine
IIR2.ASM	IIR (cascade sections, modified direct form) filter routine
IIR3.ASM	IIR (cascade sections, balanced pole-zero structure) filter routine
XFER.ASM	complex-ratio routine used for transfer function
ASM.BAT	batch file used to assemble any source module: type "ASM filename" (no extension needed)
LNK.BAT	batch file used to link all source modules: type "LNK"
T30LINK.CMD	link control file (command file for linker)

The distribution disk does not initially contain all .OBJ files necessary to produce a complete baseline, default executable code module when the T30LINK.CMD control file (above) is invoked with the Texas Instruments LNK30 utility. Each .ASM module above must be assembled first.

4.5.3 TMS320C3x Modification Process

NOTE: Be sure to save a backup copy of TMS32030.OUT before running the "LNK" batch file in case your new code doesn't work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to TMS320C3x source code, keeping in mind the necessary shared variables between Hypersignal and the TMS320C3x (the T30DEF.ASM module is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .OBJ (object) file is generated for each altered module. For convenience, an "ASM.BAT" batch file has been provided that you may use in the form of:

ASM filename <car ret>

to assemble source code modules (entering an ".ASM" extension after "filename" is optional). This file executes a

asm30 -L %1

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the T30LINK.CMD control file (use an ASCII text-editor to update the T30LINK.CMD file).

3) Re-link all TMS320C3x modules using the Texas Instruments LNK30 linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .OBJ files. LNK.BAT executes a

lnk30 T30LINK.CMD

statement, where "T30LINK.CMD" is an ASCII linker-control file that specifies all .OBJ files to be linked by the LNK30.EXE program. Note that T30LINK.CMD places shared and local variables at 400H and 480H, respectively. All code has been tested using version 4.5 of the Texas Instruments TMS320C3x/C4x development tools.

4) Copy the resulting "TMS32030.OUT" file to the directory with Hypersignal software, run Hypersignal, and invoke the appropriate functions. Hypersignal reads the "COFF" file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING TMS320C3x SOURCE CODE

1) Avoid using the default ".TEXT", ".DATA", or ".BSS" sections in any new code you might add. For some reason, the Texas Instruments linker wants to put these sections in front of all named sections, unless you specifically tell it otherwise in the .CMD file. If one of these sections loads in front of the "INIT" section, which is supposed to load at zero because it contains the reset vector, your code is dead.

2) *Do not declare new variables before or in the "HOST_COMM" section (file T30DEF.ASM).* These are the host-DSP intercommunication memory areas that must remain unchanged to allow normal Hypersignal operation.

3) Note that the start of executable code used by Hypersignal is at 500H—if you declare enough variables in section "LOCAL" (file T30DEF.ASM) or immediately after it, you could overrun the start address. If necessary put the variables somewhere else or change the address of "INIT" (both in T30BEG.ASM and T30LINK.CMD). Also note that analog and acceleration processing data buffers start at address 2000H. If you add enough code, you could overrun this address. If so, change the buffer address value "ANABFK" only—Hypersignal will base all other buffers on this value. *If you need to move any buffer addresses, please see the T30DEF.ASM file—it contains memory maps and numerous suggestions on changing default addresses.*

4) For the ASPI board, any variables between 400H and 500H (normally includes all shared and local variables) are translated to dual-port memory area 800400H-8004FFH by the Hypersignal high-level downloader, and the "VARS" location is written as 80H. For the LSI board the 400H-4FFH area is translated to dual-port memory area 30400H, and the "VARS" location is written as 3. These relocations are done so that shared variables can be accessed by the host without putting the processor in hold.

5) For the LSI board, data buffers are also moved to the dual-port memory area (instead of at 2000H, in the "program" memory area). See the T30DEF.ASM module for details on the exact translation that occurs and under what circumstances.

6) The LNK.BAT file produces a "T30LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

7) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

SPECIAL NOTES FOR THE TEXAS INSTRUMENTS EVM BOARD AND DOMAIN TECHNOLOGIES DSPCARD

The Texas Instruments EVM board and Domain Technologies DSPCard are special cases. For both these boards, the .ASM files EVM_INIT, GET_COMM, RD_DATA, and WR_DATA can be assembled and linked (using the EVMLNK.BAT and DOMLNK.BAT files) to produce the small "host interface" executable programs "EVM_INIT.OUT" and "DOM_INIT.OUT". These files must be downloaded and run by the Texas Instruments EVMLOAD program or Domain Technologies LDR_C30 program, respectively, prior to running Hypersignal. You can use DOS commands of the following form:

EVMLOAD EVM_INIT.OUT <car ret>

or

LDR_C30 DOM_INIT.OUT <car ret>

The EVMLOAD.EXE file should be included with your EVM board from Texas Instruments. *You must enter the ".OUT" extension when using the EVMLOAD program. Also, version 1.51 and above of EVMLOAD requires the two files "RESVCT.001" and "RESVCT.002" to be present in order to run—otherwise it spits up an "unable to initialize emulator" message.* A default "EVM_INIT.OUT" file is included on Hypersignal distribution diskette #2. Once loaded, the host interface provides a mechanism for Hypersignal to download further code and real-time data, as necessary for standard operation.

The Domain LDR_C30.EXE program can be found on the Domain "Development System for Texas Instruments 320C3x" disk. A default DOM_INIT.OUT file is included on Hypersignal distribution diskette #2.

The reason for separate host-interface files (as opposed to including the host interface section inside the standard TMS32030.OUT file) is that Texas Instruments apparently regards the downloading of code directly through the Test Bus Controller (TBC) embedded-emulation interface on the EVM board as proprietary (even though we asked for this information, they declined to provide it). Obviously, the use of separate .OUT files for programs that must coexist (and execute) at the same time can create problems. Below is some discussion of this situation and things to watch out for should you desire to use the Hypersignal DSP Source Code Interface to change TMS320C3x source code for either EVM or DSPCard operation.

The EVM_INIT.OUT and DOM_INIT.OUT code files expect to start execution at 300H by the. The area from 300H to 400H should be represented by an "uninitialized section" (as it is in the default Hypersignal TMS32030.OUT file), preventing the Hypersignal COFF file loader (or loaders from other vendors) from overwriting the EVM or DSPCard host interfaces, once present in memory and running.

When Hypersignal puts the EVM board or DSPCard in "reset", what it is actually doing is changing the reset vector to point to an idle address, where the TMS320C30 or TMS320C31 spins while still allowing the interrupt-driven host interface to function. When Hypersignal "runs" the EVM board or DSPCard, the reset vector is changed again to allow normal or user-defined functions to execute (using the reset vector in the TMS32030.OUT file). Hypersignal never puts the EVM TMS320C30 or DSPCard TMS320C31 chips in hold, since to do so would not allow any data transfer while in the hold state.

Should you get in to the technical details of it, you will find that the Hypersignal version of the EVM host interface is much less complicated than the example interface provided by Texas Instruments. No elaborate command formatting or parameter area is required. Since commands can be byte-wide, the Hypersignal interface simply sends the command type, instead of a "command start" bit followed by parameters and a "command end" bit. In the Hypersignal interface, the starting address is simply a pre-defined data transfer of one command type; other command types delineate the basic transfer mode (read 16-bit data, read 32-bit data, write 16-bit data, etc.). The count parameter has been eliminated completely since the EVM host interface always auto-increments its transfer address, and the host software always controls the number of transfers.

ACCESS AND CONTROL OF USER-DEFINED TMS320C3x SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as "new.out", after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For TMS320C3x boards, you should avoid using the program memory areas between 0CH and 0C0H, which is reserved by Texas Instruments, between 0C0H and 400H, which is often used for debuggers and monitors (for Texas Instruments EVM boards, Hypersignal uses 300H to 3FFH for a host interace), between 400H and 47FH, which is used by Hypersignal to store host-DSP interface variables and structures, and between 480H and 4FFH, which is used by Hypersignal TMS320C3x DSP code to store local variables.

4.6 NEC 77220 DSP Source Code Interface

4.6.1 77220 Overview

The 77220 Source Code Interface distribution diskette contains all NEC 77220 source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic NEC 77220 source code includes interrupt service routines for the NEC DDK77220 board.

As is, the 77220 source code includes both library routines and also real-time interrupt driven “modes” of operation that use foreground processing and interrupt service routine processing to implement standard Hypersignal functions. These functions have been written in modular form, and support the basic “modes” of operation below:

77220 Module	Hypersignal Function	oper. “mode”
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

MODIFICATION OF EXISTING SOURCE CODE

If the user wishes to alter the performance of the code, he/she must keep in mind the necessary links between Hypersignal software and the 77220 code. The two-way communication between the PC and 77220 established in the Hypersignal software must not be changed, as the high-level software expects to read/write certain memory locations. These memory locations are initialized by the high-level software before any code executes on the DSP board (some locations are actually read by the high-level software as well, allowing you to change such “constants” as buffer base addresses). *Do not insert any code of any type that will disturb these addresses.* The 77220 source code has been documented to aid the user in these areas.

FOREGROUND PROCESSING AND INTERRUPT STRUCTURE

The 77220 only has one external interrupt (except for NMI) which is currently used to implement a ‘pseudo-hold’ function to facilitate communication with the host PC (see MISCELLANEOUS NOTES, below). Thus, for the DDK 77220 board imple-

mentation, all analog I/O is performed via polling of the onchip serial port status, requiring that all analog I/O data processing be done in the "foreground", on a buffer-loop basis. Since further processing on buffers must begin as soon as they are full, this means that if the processing to be done is significant (which it is for the Spectrum Analyzer function), then the next data buffer won't be continuous, because no further samples can be acquired (without a sample interrupt) until the current buffer is completely processed.

4.6.2 77220 Source and Support File List

Source code and related files included on the distribution diskette:

N20DEF.SRC	variable & constant definition module
N20BEG.SRC	initialization, including ISR vectors
MODUL0.SRC	foreground A/D to disk processing (mode=0)
MODUL1.SRC	foreground D/A from disk processing (mode=1)
MODUL2.SRC	foreground Digital Oscilloscope processing (mode=2)
MODUL3.SRC	FFT file-based acceleration processing (mode=3)
MODUL4.SRC	iFFT file-based acceleration processing (mode=4)
MODUL5.SRC	foreground Spectrum Analyzer process (mode=5)
MODUL6.SRC	foreground stimulus & response processing (mode=6)
ARCTAN.SRC	arctangent routine
DIV.SRC	integer-divide routine
FFT.SRC	variable N ($N=2^m$) FFT routine
FFTOUT.SRC	post-FFT processing (mag, phase, etc.)
LOG.SRC	log approximation routine
MAG.SRC	magnitude routine (complex input)
MAX.SRC	find maximum-in-buffer routine
MIN.SRC	find minimum-in-buffer routine
PHASE.SRC	4-quadrant phase routine ($-\pi$ to $+\pi$; complex input)
SINTAB.SRC	sine value lookup-table for FFT use
SPAOUT.SRC	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.SRC	squareroot routine
WINDOW.SRC	apply pre-defined window (used with FFTs)
FILTER.SRC	filter control routine for real-time filtering
FIR.SRC	FIR filter routine
IIR2.SRC	IIR (cascade sections, modified direct form) filter routine
IIR3.SRC	IIR (cascade sections, balanced pole-zero structure) filter routine
EXTRN.INC	include file containing reference to external variable definitions for use in other .SRC files (see N20DEF.SRC above)
SRC.BAT	batch file used to assemble any source module: type "SRC filename" (no extension needed)

ASM.BAT	batch file used to assemble any assembly module: type “ASM filename” (no extension needed)
LNK.BAT	batch file used to link all source modules: type “LNK”
N20LINK.CTL	link control file (command file for linker)

The distribution disk does not initially contain all .REL files necessary to produce a complete baseline, default executable code module when the N20LINK.CTL control file (above) is invoked with the NEC LK77230 utility. Each .SRC module above must be assembled first.

4.6.3 77220 Modification Process

NOTE: Be sure to save a backup copy of NEC77220.HEX before running the “LNK” batch file in case your new code doesn’t work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to 77220 source code, keeping in mind the necessary shared variables between Hypersignal and the 77220 (the N20DEF.SRC module is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .OBJ (object) file is generated for each module. For convenience, a “SRC.BAT” batch file has been provided that you may use in the form of:

SRC filename <car ret>

to assemble source code modules (an “.SRC” extension after “filename” is optional). This file executes a

ra77230 nopre=%1 cp=20 object=%1 pri=%1.LST

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the N20LINK.CTL control file (use an ASCII text-editor to update the N20LINK.CTL file).

3) Re-link all 77220 modules using the NEC 77220 linker. A “LNK.BAT” batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .LNK files. LNK.BAT executes the following statements:

lk77230 !N20LINK.CTL oc77230 NEC77220.LNK to NEC77220.TMP hibyt220
NEC77220.TMP NEC77220.HEX del NEC77220.TMP

where “N20LINK.CTL” is an ASCII linker-control file that specifies all .LNK files to be linked by the LK77230.EXE program. Note that N20LINK.CTL blocks out (reserves) memory from 1000H to 13FFH for interrupt handling, shared and local variables, and initialization code. Thus, all relocatable code starts at 1400H. All code has been tested using version 2.0 of the NEC 77230/77220 development tools.

4) Copy the resulting “NEC77220.HEX” file to directory with Hypersignal software, run Hypersignal, and use appropriate functions. Hypersignal reads the “Intel hex” file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING 77220 SOURCE CODE

1) Although hard-core NEC 77220 programmers might say that the 77220 assembly language (“.ASM” files) is the only way to go, our programmers, with a great deal of experience in programming many other DSP devices, felt that the “pre-assembly” language (“.SRC” files) was much more like other DSP assembly languages and thus made creation of the 77220 source code more standard and easy-to-read. We have found the pre-assembler to do an excellent job in organizing code to take advantage of pipeline “delay tricks” in the 77220, as well as reliable in ensuring that either a NOP or non-related instruction is inserted when the

pre-assembly source calls for use of a result or register too early. Since the pre-assembler can output its equivalent ".ASM" file at any time for manual optimization by the user, we feel that the development time saved by the pre-assembler more than justifies its use. Thus, all Hypersignal 77220 routines are ".SRC" files.

2) Note that the "N20LINK.CTL" file (command file for linker) blocks out fast instruction memory from 0 to 3FFH. This is because these areas are occupied by absolute sections: "RESET" at 0, "INT" (interrupt service routine) at 100H, "SHARED_VARS" and "LOCAL_VARS" at 200H and 280H, respectively, and "INIT" at 300H. Thus, all relocatable modules start at 400H, and are organized into successive chunks as long as necessary by the order in which they appear in the ".CTL" file.

3) *Do not declare new variables before or in the "SHARED_VARS" section (file N20DEF.SRC) starting at 200H in fast instruction memory.* These are the shared host-DSP intercommunication memory areas that must remain unchanged to allow normal Hypersignal operation. To add memory variables, we suggest adding them after existing variables in the "LOCAL_VARS" section (also file N20DEF.SRC) starting at 280H. Note that these sections reserve memory from 200H to 2FFH in fast instruction memory, and that there is code both before and after this section, so be careful of adding too many variables.

4) The LNK.BAT file produces a "N20LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

5) The "pseudo-hold" interrupt routine at 100H responds to a "HOLD" interrupt from the host PC, saves any of the current machine state that might be modified, and calls a routine in internal instruction EPROM that waits until the host PC says it is "free to go." This effectively puts the 77220 in a hold state, allowing the host PC to access the external data/instruction memory on the board. This mechanism allows the high-level Hypersignal software to operate the DDK 77220 board in a real-time mode, transferring analog I/O data continuously when necessary, and also allows the high-level software and the low-level 77220 code to operate asynchronously using "message passing" without the need to reset the 77220 or synchronize using I/O register flags.

6) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED 77220 SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as "new.out", after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

4.7 Analog Devices ADSP-210x DSP Source Code Interface

4.7.1 ADSP-210x Overview

The ADSP-21xx Source Code Interface distribution diskette contains all Analog Devices ADSP-210x source code for ADSP-2100 and ADSP-2101 boards, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic ADSP-2101 source code includes interrupt service routines for the LSI/Spectrum 2100 and 2101 system boards.

As is, the ADSP-210x source code includes both library routines and also real-time interrupt driven "modes" of operation that use foreground processing and interrupt service routine processing to implement standard Hypersignal functions. These functions have been written in modular form, and support the basic "modes" of operation below:

210x Module	Hypersignal Function	oper. "mode"
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

4.7.2 ADSP-210x Source and Support File List

Source code and related files included on the distribution diskette:

A21DEF.DSP	variable & constant definition module
A21BEG.DSP	power-on/reset initialization
A21INT.DSP	interrupt and reset vector definition
ISR0.DSP	A/D to disk ISR for LSI/Spectrum system board
ISR1.DSP	D/A from disk ISR for LSI/Spectrum system board
ISR2.DSP	Digital Oscilloscope ISR for LSI/Spectrum system board
ISR5.DSP	Spectrum Analyzer ISR for LSI/Spectrum system board
ISR6.DSP	Stimulus & response ISR for LSI/Spectrum system board
MODUL0.DSP	foreground A/D to disk processing (mode=0)
MODUL1.DSP	foreground D/A from disk processing (mode=1)
MODUL2.DSP	foreground digital scope processing (mode=2)

MODUL3.DSP	FFT file-based acceleration processing (mode=3)
MODUL4.DSP	iFFT file-based acceleration processing (mode=4)
MODUL5.DSP	foreground Spectrum Analyzer process (mode=5)
MODUL6.DSP	foreground stimulus & response process (mode=6)
MODUL7.DSP	convolution/correlation file-based acceleration (mode=7)
ARCTAN.DSP	arctangent routine
BITREV.DSP	bit-reverse routine
CONVLV.DSP	convolution/correlation routine
DIVIDE.DSP	fractional and integer divide routines
FFT.DSP	variable N ($N=2^m$) FFT routine
FFTOUT.DSP	post-FFT processing (mag, phase, etc.)
LOG.DSP	log approximation routine
MAG.DSP	magnitude routine (complex input)
MAX.DSP	find maximum-in-buffer routine
MIN.DSP	find minimum-in-buffer routine
PHASE.DSP	4-quadrant phase routine (complex input)
SINTAB.DSP	sine value lookup-table for FFT use
SPAOUT.DSP	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.DSP	squareroot routine
WINDOW.DSP	apply pre-defined window (used with FFTs)
FILTER.DSP	filter control routine for real-time filtering
FIR.DSP	FIR filter routine
IIR2.DSP	IIR (cascade sections, modified direct form) filter routine
IIR3.DSP	IIR (cascade sections, balanced pole-zero structure) filter routine
XFER.DSP	complex-ratio routine used for transfer function
ASM.BAT	batch file used to assemble any source module: type “ASM filename” (no extension needed)
LNK.BAT	batch file used to link all source modules: type “LNK”
A21LINK.CTL	link control file (command file for linker)
HSMACRO.SYS	“system builder” file
HSMACRO.ACH	“system architecture” file

The distribution disk does not initially contain all .OBJ, .CDE, and .INT files necessary to produce a complete baseline, default executable code module when the A21LINK.CTL control file (above) is invoked with the Analog Devices LD21 link utility. Each .DSP module above must be assembled first.

4.7.3 ADSP-210x Modification Process

NOTE: Be sure to save a backup copy of ADSP210x.EXE before running the "LNK" batch file in case your new code doesn't work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to ADSP-210x source code, keeping in mind the necessary shared variables between Hypersignal and the ADSP-210x (the A21DEF.DSP module is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .OBJ (object), .CDE, and .INT files are generated for each altered module. For convenience, an "ASM.BAT" batch file has been provided that you may use in the form of:

ASM filename <car ret>

to assemble source code modules (entering a ".DSP" extension after "filename" is optional). This file executes a

ASM21 -L %1

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the A21LINK.CTL control file (use an ASCII text-editor to update the A21LINK.CTL file).

3) Re-link all ADSP-210x modules using the Analog Devices LD21 linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .OBJ, .CDE, and .INT files. LNK.BAT executes the following statements:

LD21 A21LINK.CTL -a HSMACRO -e AD210x -g -x DEL A21LINK.MAP REN AD210x.MAP A21LINK.MAP

where "A21LINK.CTL" is an ASCII linker-control file that specifies all .OBJ, .CDE, and .INT files to be linked by the LD21 link utility, and "A21LINK.MAP" is a map file showing variable and module addresses and lengths (see notes below). Note that A21LINK.CTL places shared and local variables at 001CH, forcing relocatable code to start after these declarations. All code has been tested using version 1.13 (11/21/90) of the Analog Devices ASM21 assembler, and version 1.12 (11/15/90) of the Analog Devices LD21 linker.

4) Copy the resulting "ADSP210x.EXE" file to the directory with Hypersignal software, run Hypersignal, and invoke the appropriate functions. Hypersignal reads the Analog Devices .EXE file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING ADSP-210x SOURCE CODE

1) The basic LSI ADSP-2100 and ADSP-210x boards support only 4kX24 of program memory. Thus, you must be careful when adding program memory variables or code so that overall code space is not exceeded. Program memory can be expanded to 16kX24.

2) Because of the 4kX24 minimum program memory size, the look-up table used by the Hypersignal FFT routine, which is stored in program memory, has been limited to a 1024-pt. FFT size. Because the FFT routine used requires the table to be in bit-reversed order, both cosine and sine values are required. Unfortunately, the ADSP-210x bit-reversed addressing facility is not very flexible, not only placing restrictions on where such values are located in memory, but also creating the actual bit-reversed address only in hardware without making it available in software (via register). Should you desire to increase the FFT size, you must obtain expanded memory and also create a new bit-reversed look-up table.

3) In its factory configuration, the LSI ADSP-2101 board supports only 4kX16 of data memory. Data memory can be expanded, but not to 16kX16 because the 2101 only provides 14 external address lines, and in addition, uses the upper 4kX16 of data memory space for onchip internal RAM and other chip peripherals (serial ports, etc.). Thus, the maximum available data memory size for standard Hypersignal purposes is 12kX16.

4) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". The value is in Hz (Hertz); note that for ADSP-210x boards, which use 16-bit memory accesses, the maximum value that can be represented is limited to 65535. This value is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

5) The LNK.BAT file produces a "A21LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

ACCESS AND CONTROL OF USER-DEFINED ADSP-2101 SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as "new.out", after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For ADSP-2101 boards, you should avoid using the program memory areas between 1CH and 07FH, which is used by Hypersignal to store host-DSP interface variables and structures, and between 80H and 8FH, which is used by Hypersignal ADSP-2101 DSP code to store local variables. Note also that if Hypersignal ADSP-2101 DSP code is allowed to run, filter coefficients may be stored in onchip RAM at location 3C00H.

4.8 Motorola DSP96002 DSP Source Code Interface

4.8.1 DSP96002 Overview

The DSP96002 Source Code Interface distribution diskette contains all Motorola DSP96002 source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic DSP96002 source code includes interrupt service routines for the LSI/Spectrum and Ariel DSP-96 boards.

As is, the DSP96002 source code is written in modular form, and supports the basic "modes" of operation below:

'96002 Module	Hypersignal Function	oper. "mode"
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the DSP96002 board and generally match up with the foreground processing modules above. Interrupt service routine module names consist of "ISRn", where n indicates the operating mode (see above) that the routine supports, for example "ISR0" for operating mode zero. Inside each interrupt service routine there are different entry points that contain hardware-specific input/output code for individual boards. These hardware-specific sections are kept as small as possible, so that the great majority of the routine is common for all boards. Examine the interrupt vector and mask initialization in the "M96DEF" (non-executable definitions), "M96VEC", and "M96BEG" modules, depending on board type (see host-shared variable "BOARD"), to locate the correct interrupt service routine for a particular function.

MODIFICATION OF EXISTING SOURCE CODE

If the user wishes to alter or add code functionality, or improve the functionality and/or performance of existing code, he/she must keep in mind the necessary intercommunication variables between Hypersignal software and DSP96002 code. The two-way communication between the host-PC and DSP96002 set up in the Hypersignal software must not be changed, as the software expects to read/write certain memory locations. These memory locations reside at 100H and are clearly commented inside M96DEF. *Do not insert any code of any type that would change the current addresses of these variables.* If necessary, you may add further communication variables for your own high-level code. *You may change* the values of ANABFK, WNDBFK, RIBUFK, and MPBUFK (time data buffer, FFT window data buffer, real/imaginary data buffers, & magnitude/phase data buffers, respectively) if it becomes necessary to alter memory allocation in your DSP96002 code and communicate these changes to the host PC). The DSP96002 source code has been well-documented to aid the user in these areas.

FOREGROUND PROCESSING AND REAL-TIME INTERRUPT STRUCTURE

Currently, operating mode 5 (Spectrum Analyzer) is the best example of full implementation of foreground processing with complete context-save background interrupts, although all ISRs save all registers temporarily used. The Spectrum Analyzer source code represents a totally real-time system, with samples buffered in a dual-buffer scheme in response to background "sample ready" interrupts, and buffers, when full, sent to FFT and post-FFT processing in the foreground.

Use mode 5 source modules ("MODUL5" and modules it calls), and "ISR5", for a good example of a real-time, interrupt-driven structure.

4.8.2 DSP96002 Source and Support File List

Source code and related files included on the distribution diskette:

M96DEF.ASM	variable & constant definition module
M96VEC.ASM	beginning vectors, including ISR vector initialization, host-interface definition, and power-on/reset chip initialization
M96BEG.ASM	software initialization
ISR0.ASM	A/D ISR for all boards
ISR1.ASM	D/A ISR for all boards
ISR2.ASM	Digital Oscilloscope ISR for all boards
ISR5.ASM	Spectrum Analyzer ISR for all boards
ISR6.ASM	stimulus & response ISR for all boards
MODUL0.ASM	foreground A/D processing (mode=0)
MODUL1.ASM	foreground D/A processing (mode=1)
MODUL2.ASM	foreground Digital Oscilloscope processing (mode=2)
MODUL3.ASM	FFT file-based acceleration processing (mode=3)
MODUL4.ASM	iFFT file-based acceleration processing (mode=4)
MODUL5.ASM	foreground Spectrum Analyzer processing (mode=5)
MODUL6.ASM	foreground stimulus & response processing (mode=6)
MODUL7.ASM	convolution/correlation file-based acceleration processing (mode=7)
ARCTAN.ASM	arctangent routine
CONVLV.ASM	convolution/correlation routine
DIVIDE.ASM	fractional and integer divide routines
DMA_PREP.ASM	DMA transfer preparation routine (see notes below)
FFT.ASM	FFT/inverse FFT routine
FFTOUT.ASM	post-FFT processing (mag, phase, scaling)
FILTER.ASM	filter control routine for real-time filtering
FIR.ASM	FIR filter routine
IIR2.ASM	IIR (cascade sections, modified direct form) filter routine
IIR3.ASM	IIR (cascade sections, balanced pole-zero form) filter routine
LOG.ASM	log approximation routine
MAG.ASM	magnitude routine (complex input)
PHASE.ASM	4-quadrant phase routine (complex input)
SPAOUT.ASM	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.ASM	squareroot routine
WINDOW.ASM	window routine: applies PC-downloaded window for Spectrum Analyzer and FFT
XFER.ASM	transfer function calculation (complex input)
XMAX.ASM	location of maximum value in X:buffer data

XMIN.ASM	location of minimum value in X:buffer data
YMAX.ASM	location of maximum value in Y:buffer data
XXMOVE.ASM YYMOVE.ASM XYMOVE.ASM YXMOVE.ASM LMOVE.ASM	move routines between X: and Y: memory
INTIOEQU.INC	include file containing equates for internal I/O addresses
EXTIOEQU.INC	include file containing equates for external I/O addresses
M96LINK.CTL	link control file (control file for linker)
ASM.BAT	batch file that can be used to assemble individual source code modules (use in form "ASM filename")
LNK.BAT	batch file that performs linking (use in form "LNK")

The distribution disk does not initially contain all .CLN files necessary to produce a complete baseline, default executable code module when the M96LINK.CTL control file (above) is invoked with the Motorola DSPLNK utility. Each .ASM module above must be assembled first.

4.8.3 DSP96002 Modification Process

NOTE: Be sure to save a backup copy of DSP96002.OUT before running the "LNK" batch file in case your new code doesn't work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to DSP96002 source code, keeping in mind the necessary links between Hypersignal and the DSP96002 (see discussion above, and M96DEF.ASM module, which is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .CLN (object) file is generated for each altered module. For convenience, an "ASM.BAT" batch file has been provided that you may use in the form of:

ASM filename <car ret>

to assemble source code modules (entering an ".ASM" extension after "filename" is optional). This file executes a

asm96000 -B -L %1

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the M96LINK.CTL control file (use an ASCII text-editor to update the M96LINK.CTL file).

3) Re-link all DSP96002 modules using the Motorola DSPLNK linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .LNK files. LNK.BAT executes a

DSPLNK -BDSP96002.OUT -MM96LINK.MAP -FM96LINK.CTL

statement, where "M96LINK.CTL" is an ASCII linker-control file that specifies all .CLN files to be linked by the DSPLNK.EXE program. Note that re-locatable code is positioned in internal PRAM between shared and local variables starting at 100H and software initialization code (M96BEG) starting at 400H (start of external PRAM), and afer M96BEG in external PRAM. This leaves significant space in internal PRAM for the linker to place interrupt service routines. You should not change absolute start of M96BEG at 400H unless you have a specific internal PRAM memory mapping strategy in mind; and you should not change the 100H address of shared variables for any reason, ever (as the host PC expects them there). All code has been tested using version 4.1 of ASM96000 and DSPLNK.

4) Copy the resulting "DSP96002.OUT" file to directory with Hypersignal, re-invoke the Hypersignal software, and run the appropriate function(s). Hypersignal reads the "COFF" file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING DSP96002 SOURCE CODE

1) If you need to change default buffer addresses, please consult the comments and suggestions in the M96DEF.ASM file.

2) The LNK.BAT file produces a "M96LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

3) The local variable "SEG" is used to store the current code segment page (upper 8 bits of linear address), and varies from board to board. Since virtually all DSP96002 program control instructions are not relocatable (instead using absolute addresses in the instruction sequence), all DSP96002 code must be dynamically relocated at run-time. After booting from internal PRAM, the DSP96002 code can then discover and set for future reference its code segment by examining instructions located in external PRAM.

4) In order to achieve the fastest possible means of transferring large amounts of continuous data in real-time between the host PC and the DSP96002, the Hypersignal-Macro implementation makes extensive use of the following combinations of transfer types, depending on the board in use:

a. onchip DMA between dual-port memory and other DSP96002 memory

b. onchip DMA between the host-interface and other DSP96002 memory

c. host-interface command-driven transfers

d. OnCE port driven transfers

It may be fair to say this is the most complex DSP chip interface supported by Hypersignal-Macro, of over 8 different DSP chips supported. It is definitely accurate to say that this is one of the most interesting and fascinating.

5) The LSI DSP96002 board uses IRQB for analog sample interrupts, and IRQC to initiate on-chip DMA controller data transfers. IRQA is unused. The Ariel DSP-96 board uses IRQA for analog sample interrupts, and IRQB to initiate on-chip DMA controller data transfers. IRQC is left available for use with the "heartbeat" clock interrupt on the DSP-96.

6) In no case does the Hypersignal-Macro DSP96002 source code use a dedicated register set for its host-interface definition. All registers used are saved temporarily on the stack, allowing use of a 'C' or other high-level compiler.

7) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED DSP96002 SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as “new.out”, after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For DSP96002 boards, you should avoid using the program memory areas between 1CH and 03FH, which is used by Hypersignal to implement a host-interface, between 40H and 0FFH, which is used by Hypersignal to implement setup and initialization code necessary to coordinate execution and DMA memory-mapped interfaces with the host, between 100H and 17FH, which is used by Hypersignal to store host-DSP interface variables and structures, and between 180H and 1FFH, which is used by Hypersignal DSP96002 DSP code to store local variables. Note that you can take advantage of the start up code located between by 40H and 0FFH by locating the start of your executable code at 400H, which is the initial jump taken by Hypersignal DSP96002 code after all setup and communication initialization is complete.

4.9 AT&T WE DSP3210 DSP Source Code Interface

4.9.1 WE DSP3210 Overview

The DSP3210 Source Code Interface distribution diskette contains all AT&T WE DSP3210 source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic DSP3210 source code includes interrupt service routines for the Ariel DSP-3210 board and CAC QUANTUMdsp board.

As is, the DSP3210 source code is written in modular form, and supports the basic “modes” of operation below:

‘3210 Module	Hypersignal Function	oper. “mode”
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the DSP3210 boards and generally match up with the foreground processing modules above. Interrupt service routine module names consist of “ISRn”, where n indicates the operating mode (see above) that the routine supports, for example “ISR0" for operating mode zero. Inside each interrupt service routine there are different entry points that contain hardware-specific input/output code for individual boards. These

hardware-specific sections are kept as small as possible, so that the great majority of the routine is common for all boards. Examine the interrupt vector and mask initialization (”itable" and PCW register) in RESET.S and W32BEG.S to locate the correct interrupt service routine for a particular function.

4.9.2 WE DSP3210 Source and Support File List

Source code and related files included on distribution diskette:

RESET.S	reset vector, wait-state, and “page register” initialization
W32DEF.S	variable & constant definition module
W32BEG.S	initialization, including ISR vectors
ISR0.S	A/D to disk ISR for Ariel DSP-3210 board
ISR1.S	D/A from disk ISR for all boards
ISR2.S	Digital Oscillscope ISR for all boards
ISR5.S	Spectrum Analyzer ISR for all boards
ISR6.S	Stimulus & response ISR for all boards
ISR6.S	Stimulus & response ISR for boards
ARI_TX.S	analog I/O output ISR for Ariel DSP-3210 only
MODUL0.S	foreground A/D to disk processing (mode=0)
MODUL1.S	foreground D/A from disk processing (mode=1)
MODUL2.S	foreground Digital Oscilloscope processing (mode=2)
MODUL3.S	FFT file-based acceleration processing (mode=3)
MODUL4.S	iFFT file-based acceleration processing (mode=4)
MODUL5.S	foreground Spectrum Analyzer process (mode=5)
MODUL6.S	foreground stim & response process (mode=6)
MODUL7.S	convolution/correlation file-based acceleration (mode=7)
ARCTAN.S	arctangent routine
CONVLV.S	convolution/correlation routine
DIV.S	divide routine
FFT.S	variable N (N=2^FFT_order) FFT routine
FFTOUT.S	post-FFT processing (mag, phase, etc.)
FILTER.S	filter control routine for real-time filtering
FIR.S	FIR filter routine
IIR2.S	IIR (cascade sections, modified direct form) filter routine
IIR3.S	IIR (cascade sections, balanced pole-zero structure) filter routine
LOG10.S	log (base 10) approximation routine
MAXFIND.S	find maximum-in-buffer routine
MINFIND.S	find minimum-in-buffer routine
MAG.S	magnitude routine (complex input)
PHASE.S	4-quadrant phase routine ($-\pi$ to $+\pi$; complex input)

SPAOUT.S	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.S	squareroot routine
WINDOW.S	apply pre-defined window (used with FFTs)
XFER.S	complex-ratio routine used for transfer function
ASM.BAT	batch file used to assemble any source module: type “ASM filename” (no extension needed)
LNK.BAT	batch file used to link all source modules: type “LNK”
W32LINK.CTL	link control file (command file for linker)

The distribution disk does not initially contain all .O files necessary to produce a complete baseline, default executable code module when the W32LINK.CTL control file (above) is invoked with the AT&T d32ld utility. Each .S module above must be assembled first.

4.9.3 WE DSP3210 Modification Process

NOTE: Be sure to save a backup copy of DSP3210.OUT before running the “LNK” batch file in case your new code doesn’t work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to DSP3210 source code, keeping in mind the necessary shared variable links between Hypersignal and the DSP3210 (see discussion above and W32DEF.S module, which is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .O (object) file is generated for each altered module. For convenience, an “ASM.BAT” batch file has been provided that you may use in the form of:

ASM filename <car ret>

to assemble source code modules (*do not enter a “.S” extension after “filename”*). This file executes a

d32as -b l -lo %1 %1.s

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the W32LINK.CTL control file (use an ASCII text-editor to update the W32LINK.CTL file).

3) Re-link all DSP3210 modules using the AT&T DL3D linker. A “LNK.BAT” batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .O files. LNK.BAT executes the following statements:

```
D32LD -abs -map -o DSP3210.OUT W32LINK.CTL > W32LINK.MAP
@COPY W32LINK.MAP+EOF W32LINK.MAP
@D32NM -f DSP3210.OUT > TEMP
@COPY W32LINK.MAP+TEMP W32LINK.MAP
@DEL TEMP
```

where “W32LINK.CTL” is an ASCII linker-control file that specifies all .O files to be linked by the d32ld.exe program, and W32LINK.MAP is a map file showing variable and module addresses and lengths (W32LINK.MAP is created using DOS file-redirection commands, since the AT&T linker does not support a map file output option directly; see notes below). Note that W32LINK.CTL places the reset vector module at zero, and shared variables at 0x100, forcing all relocatable code to start after the variable declarations. All code has been tested using version 1.2 of d32as and d32ld.

4) Copy the "DSP3210.OUT" file to the directory with the Hypersignal HSxxxxx.OVR and HSxxxxx.EXE executable files, run Hypersignal, and invoke the appropriate functions. Hypersignal reads the ".OUT" file (which is in "COFF" format) directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING DSP3210 SOURCE CODE

1) Hypersignal DSP3210 code uses r18 as a system stack pointer for both foreground and interrupt processing code. When storing CAU registers on the "stack", use the following sequence:

```
*r18++ = rN1
*r18++ = rN2
        -
        -
        -
*r18++ = rNN
```

When restoring CAU registers from the stack, use the following sequence:

```
rN1 = r18 - 4
rNN = *rN1—
  -
  -
  -
rN2 = *rN1—
rN1 = *rN1
r18 = r18 - K        /* K = 4 x (number of registers stored) */
```

There are three notable things about this process. First, the stack is always expected to increment and decrement in 4-byte chunks (always save/restore extended registers). Second, the stack is always expected to point to the *next available (free) location*. Third, when restoring, you can't simply do an r18e = r18 - 4 instruction and begin popping rXX = *r18— instructions, because an interrupt might occur sometime after the r18 - 4 instruction and sometime before you finish popping; the interrupt routine, if it also used r18 as a stack pointer (which all types of routines should be able to do of a legitimate stack pointer), might then clobber a stored register (or more than one) that hadn't been popped yet.

2) Hypersignal DSP3210 source code is built with the "ASM" and "LNK" batch files, which invoke the AT&T "d32as" and "d32ld" assembler and linker utilities, respectively. By examining the ASM and LNK batch files, you should be able to construct an equivalent method of building the code using the AT&T "make" utility instead; however, producing a memory map file will be difficult. For the DSP Source Code Interface, obscure DOS commands have been used to create a map file in the LNK.BAT file, since the AT&T utilities will only print map output to the screen. The map file produced has a filename of "W32LINK.MAP"; you can use this file to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

3) In addition to setting them directly in the DSP3210 source code, IOC register settings can be controlled from the high-level software in order to accommodate non-standard analog I/O connected to DSP3210 serial ports. Inside the Hypersignal-Macro configuration file there is a 32-bit integer IOC value for every DSP3210 board currently supported by Hypersignal-Macro—see the "CONFIG.DOC" file on disk #2 for a list of the sequence of variables in the configuration file. Depending on the board currently in use, one of these values is sent down to the DSP3210 board whenever a Hypersignal-Macro real-time function requiring analog I/O is invoked. The corresponding DSP source code variable is "IOC_reg", which is normally used directly to set the IOC register (unless changed by the user in the DSP source code).

4) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to

the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED DSP3210 SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as "new.out", after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For DSP3210 boards, you should avoid using the area between 100H and 300H, which is used by Hypersignal to store host-DSP interface variables and structures (100H-1FFH) and local DSP variables (200H-3FFH).

4.10 Texas Instruments TMS320C3x DSP Source Code Interface

4.10.1 TMS320C4x Overview

The TMS320C4x Source Code Interface distribution diskette contains all Texas Instruments TMS320C4x source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic TMS320C4x source code includes interrupt service routines for the Ariel DSP-C40 board.

As is, the TMS320C4x source code includes both library routines and also real-time interrupt driven "modes" of operation that use foreground processing and interrupt service routine processing to implement standard Hypersignal functions. These functions have been written in modular form, and support the basic "modes" of operation below:

'C4x Module	Hypersignal Function	oper. "mode"
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4

MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the TMS320C4x boards and generally match up with the foreground processing modules above. Interrupt service routine module names consist of “ISRn”, where n indicates the operating mode (see above) that they support, for example “ISR0". Inside each interrupt service routine there are different entry points that contain hardware-specific input/output code for individual boards. These hardware-specific sections are kept as small as possible, so that the great majority of the routine is common for all boards. Examine the interrupt vector and mask initialization in the initial setup code in ”T40BEG", depending on board type (see host-shared variable “BOARD”), to locate the correct interrupt service routine for a particular function.

MODIFICATION OF EXISTING SOURCE CODE

If the user wishes to alter the performance of the code, he/she must keep in mind the necessary links between Hypersignal software and the TMS320C4x code. The two-way communication between the PC and TMS320C4x processor established in the Hypersignal software must not be changed, as the high-level software expects to read/write certain memory locations. These memory locations are initialized by the high-level software before any code executes on the DSP board (some locations are actually read by the high-level software as well, allowing you to change such “constants” as buffer base addresses). *Do not insert any code of any type before these addresses.* The TMS320C4x source code has been documented to aid the user in these areas.

FOREGROUND PROCESSING AND REAL-TIME INTERRUPT STRUCTURE

Currently, operating mode 5 (Spectrum Analyzer) is the best example of full implementation of foreground processing with complete context-save background interrupts. The Spectrum Analyzer source code represents a totally real-time system, with samples buffered in a dual-buffer scheme in response to background “sample ready” interrupts, and buffers, when full, sent to FFT and post-FFT processing in the foreground. The dual-buffer communication between the host and DSP is fully synchronized using “interlock” buffer flags. Use mode 5 source modules (“MODUL5" and routines it calls) and ”xxx_ISR5" for one example of a real-time, interrupt-driven structure.

4.10.2 TMS320C4x Source and Support File List

Source code and related files included on the distribution diskette:

T40DEF.ASM	variable & constant definition module
T40BEG.ASM	initialization, including ISR vectors
ISR0.ASM	A/D to disk ISR for all boards
ISR1.ASM	D/A from disk ISR for all boards
ISR2.ASM	Digital Scope ISR for all boards
ISR5.ASM	Spectrum Analyzer ISR for all boards
ISR6.ASM	stimulus & response ISR for all boards
MODUL0.ASM	foreground A/D to disk processing (mode=0)
MODUL1.ASM	foreground D/A from disk processing (mode=1)
MODUL2.ASM	foreground Digital Oscilloscope processing (mode=2)
MODUL3.ASM	FFT file-based acceleration processing (mode=3)
MODUL4.ASM	iFFT file-based acceleration processing (mode=4)
MODUL5.ASM	foreground Spectrum Analyzer process (mode=5)

MODUL6.ASM	foreground stimulus & response process (mode=6)
MODUL7.ASM	convolution/correlation file-based acceleration (mode=7)
ARCTAN.ASM	arctangent routine
CONVLV.ASM	convolution/correlation routine
DIVF.ASM	floating-point divide routine
FFT.ASM	variable N ($N=2^m$) FFT routine
FFTOUT.ASM	post-FFT processing (mag, phase, etc.)
LOG.ASM	log approximation routine
MAG.ASM	magnitude routine (complex input)
MAX.ASM	find maximum-in-buffer routine
MIN.ASM	find minimum-in-buffer routine
PHASE.ASM	4-quadrant phase routine ($-\pi$ to $+\pi$; complex input)
SINTAB.ASM	sine value lookup-table for FFT use
SPAOUT.ASM	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.ASM	squareroot routine
WINDOW.ASM	apply pre-defined window (used with FFTs)
FILTER.ASM	filter control routine for real-time filtering
FIR.ASM	FIR filter routine
IIR2.ASM	IIR (cascade sections, modified direct form) filter routine
IIR3.ASM	IIR (cascade sections, balanced pole-zero structure) filter routine
XFER.ASM	complex-ratio routine used for transfer function
ASM.BAT	batch file used to assemble any source module: type “ASM filename” (no extension needed)
LNK.BAT	batch file used to link all source modules: type “LNK”
T40LINK.CMD	link control file (command file for linker)

The distribution disk does not initially contain all .OBJ files necessary to produce a complete baseline, default executable code module when the T40LINK.CMD control file (above) is invoked with the Texas Instruments LNK40 utility. Each .ASM module above must be assembled first.

4.10.3 TMS320C4x Modification Process

NOTE: Be sure to save a backup copy of TMS32040.OUT before running the “LNK” batch file in case your new code doesn’t work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to TMS320C4x source code, keeping in mind the necessary shared variables between Hypersignal and the TMS320C4x (the T40DEF.ASM module is well-documented in this area).

2) Assemble all modules changed and any new modules created to ensure a new .OBJ (object) file is generated for each altered module. For convenience, an “ASM.BAT” batch file has been provided that you may use in the form of:

ASM filename <car ret>

to assemble source code modules (entering an ".ASM" extension after "filename" is optional). This file executes a

asm30 -l -v40 %1

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the T40LINK.CMD control file (use an ASCII text-editor to update the T40LINK.CMD file).

3) Re-link all TMS320C4x modules using the Texas Instruments LNK30 linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .OBJ files. LNK.BAT executes a

lnk30 -v40 T40LINK.CMD

statement, where "T40LINK.CMD" is an ASCII linker-control file that specifies all .OBJ files to be linked by the LNK30.EXE program. Note that T40LINK.CMD places shared and local variables at 400H and 480H, respectively. All code has been tested using version 4.5 of the Texas Instruments TMS320C3x/4x development tools.

4) Copy the resulting "TMS32040.OUT" file to the directory with Hypersignal software, run Hypersignal, and invoke the appropriate functions. Hypersignal reads the "COFF" file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING TMS320C4x SOURCE CODE

1) Avoid using the default ".TEXT", ".DATA", or ".BSS" sections in any new code you might add. For some reason, the Texas Instruments linker wants to put these sections in front of all named sections, unless you specifically tell it otherwise in the .CMD file. If one of these sections loads in front of the "INIT" section, which is supposed to load at zero because it contains the reset vector, your code is dead.

2) *Do not declare new variables before or in the "HOST_COMM" section (file T40DEF.ASM).* These are the host-DSP intercommunication memory areas that must remain unchanged to allow normal Hypersignal operation.

3) Note that the start of executable code used by Hypersignal is at 500H—if you declare enough variables in section "LOCAL" (file T40DEF.ASM) or immediately after it, you could overrun the start address. If necessary put the variables somewhere else or change the address of "INIT" (both in T40BEG.ASM and T40LINK.CMD). Also note that analog and acceleration processing data buffers start at address 2000H. If you add enough code, you could overrun this address. If so, change the buffer address value "ANABFK" only—Hypersignal will base all other buffers on this value. *If you need to move any buffer addresses, please see the T40DEF.ASM file—it contains memory maps and numerous suggestions on changing default addresses.*

4) The LNK.BAT file produces a "T40LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

5) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". This value is in Hz (Hertz) and is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED TMS320C4x SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as “new.out”, after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For TMS320C4x boards, you should avoid using the program memory areas between 0CH and 0C0H, which are reserved by Texas Instruments, between 0C0H and 400H, which is often used for debuggers and monitors, between 400H and 47FH, which is used by Hypersignal to store host-DSP interface variables and structures, and between 480H and 4FFH, which is used by Hypersignal TMS320C4x DSP code to store local variables.

4.11 Texas Instruments TMS320C5x DSP Source Code Interface

4.11.1 TMS320C5x Overview

The TMS320C5x Source Code Interface distribution diskette contains all TMS320C5x source code, including linker control files and support utilities. It is highly recommended that you make a backup copy of this disk before making any changes.

The basic TMS320C5x source code includes interrupt service routines for the Innovative Integration PC31 board.

As is, the TMS320C5x source code is written in modular form, and supports the basic “modes” of operation below:

‘C5x Module	Hypersignal Functions	oper. “mode”
MODUL0	Analog Conversion	mode=0
MODUL1	Analog Conversion, Playback from Waveform Display	mode=1
MODUL2	Digital Oscilloscope	mode=2
MODUL3	FFT file-based acceleration	mode=3
MODUL4	iFFT file-based acceleration	mode=4
MODUL5	Spectrum Analyzer	mode=5
MODUL6	Stimulus & Response operation	mode=6
MODUL7	Convolution/Correlation file-based acceleration	mode=7
MODULn	User-Defined function	mode=n

INTERRUPT SERVICE ROUTINES (ISRs)

There are various interrupt service routines that respond to interrupts on the TMS320C5x boards and generally match up with the foreground processing modules above. Interrupt service routine module names consist of “ISRn”, where n indicates the operating mode (see above) that they support, for example “ISR0". Inside each interrupt service routine there are different entry points that contain hardware-specific input/output code for individual boards. These hardware-specific sections are kept as small as possible, so that the great majority of the routine is common for all boards. Examine the interrupt vector and mask initialization in the initial setup code in ”T50BEG", depending on board type (see host-shared variable “BOARD”), to locate the correct interrupt service routine for a particular function.

MODIFICATION OF EXISTING SOURCE CODE

If the user wishes to alter the performance of the code, he/she must keep in mind the necessary links between Hypersignal software and the TMS320C5x code. The two-way communication between the PC and TMS320C5x established in the Hypersignal software must not be changed, as the high-level host software expects to read/write certain memory locations. These memory locations are initialized by table reads from program memory inside module T50BEG, and are suffixed by ‘ADD’, as in MODADD, TRGADD, etc. *Do not insert any code of any type before these addresses that would cause them to move.* The TMS320C5x source code has been documented to aid the user in these areas.

FOREGROUND PROCESSING AND REAL-TIME INTERRUPT STRUCTURE

Currently, operating mode 5 (Spectrum Analyzer) is the best example of full implementation of foreground processing with complete context-save background interrupts. The Spectrum Analyzer source code represents a totally real-time system, with samples buffered in a dual-buffer scheme in response to background “sample ready” interrupts, and buffers, when full, sent to FFT and post-FFT processing in the foreground. Use mode 5 source modules (“MODUL5" and modules it calls), and ”xxISR5" for one example of a real-time, interrupt-driven structure.

4.11.2 TMS320C5x Source and Support File List

Source code and related files included on the distribution diskette:

T50DEF.ASM	variable & constant definition
T50BEG.ASM	beginning vectors, including ISR vectors, and including initialization, including PC-written variables
ISR0.ASM	continuous A/D ISR for all boards
ISR1.ASM	continuous D/A ISR for all boards
ISR2.ASM	Digital Scope ISR for all boards
ISR5.ASM	Spectrum Analyzer ISR for all boards
ISR6.ASM	Stimulus & Response ISR for all boards
MODUL0.ASM	foreground A/D processing (mode=0)
MODUL1.ASM	foreground D/A processing (mode=1)
MODUL2.ASM	foreground Digital Oscilloscope processing (mode=2)
MODUL3.ASM	FFT file-based acceleration processing (mode=3)
MODUL4.ASM	iFFT file-based acceleration processing (mode=4)
MODUL5.ASM	foreground Spectrum Analyzer process (mode=5)
MODUL6.ASM	foreground stimulus & response (mode=6)
MODUL7.ASM	convolution/correlation file-based acceleration processing (mode=7)
ARCTAN.ASM	arctangent routine

CONVLV.ASM	convolution/correlation routine
DIVIDE.ASM	fractional and integer divide routines
EXP2.ASM	exponentiation (base-2) routine (needed for fixed-point transfer function calculation)
FFT.ASM	variable N ($N=2^m$) FFT routine
FFTOUT.ASM	post-FFT processing (mag, phase, scaling)
FILTER.ASM	invokes correct filter for current channel
FIR.ASM	FIR filter routine
IIR2.ASM	IIR filter cascade struc 2 (modified direct-form)
IIR3.ASM	IIR filter cascade struc 3 (balanced pole-zero)
LOG2.ASM	log2 approximation routine
MAG.ASM	magnitude routine
MAX.ASM	find maximum routine
MIN.ASM	find minimum routine
MOVE.ASM	move data
PHASE.ASM	4-quadrant phase routine
SINTAB.ASM	sine value lookup-table for FFT use
SPAOUT.ASM	post-FFT processing for Spectrum Analyzer function, including mag, phase, power-spectra, log, scaling, and data manipulation
SQRT.ASM	squareroot routine
WINDOW.ASM	apply pre-defined window (used with FFTs)
XFER.ASM	complex-ratio routine used for transfer function
ASM.BAT	batch file used to assemble any source module: type “ASM filename” (no extension needed)
LNK.BAT	batch file used to link all source modules: type “LNK”
T50LINK.CMD	link control file (batch file for linker)

The distribution disk does not initially contain all .OBJ files necessary to produce a complete baseline, default executable code module when the T50LINK.CMD control file (above) is invoked with the Texas Instruments “dsplnk” utility. Each .ASM module above must be assembled first.

4.11.3 TMS320C5x Modification Process

NOTE: Be sure to save a backup copy of TMS32025.OUT before running the “LNK” batch file in case your new code doesn’t work.

Below are the steps taken to modify, add, and integrate new source code:

1) Make changes to TMS320C5x source code, keeping in mind the necessary shared variables between Hypersignal and the TMS320C5x. The file T50DEF.ASM is well-documented in this area.

2) Assemble all modules changed and any new modules created (using Texas Instruments “COFF” assembler) to ensure that a new .OBJ (object) file is generated for each altered module. For convenience, an “ASM.BAT” batch file has been provided that can be invoked as:

ASM filename <car ret>

to assemble source code modules (entering the ".ASM" extension after "filename" is optional). This file executes a

dspa -l -v50 -w %1

statement, where %1 is the filename.

If a new module has been created, be sure to add it to the T50LINK.CMD control file (use an ASCII text-editor to update the T50LINK.CMD file).

3) Re-link all modules using Texas Instruments TMS320C5x "dsplnk" linker. A "LNK.BAT" batch file has been provided that you may use by invoking:

LNK <car ret>

to link all necessary .OBJ files. LNK.BAT executes a

dsplnk T50LINK.CMD

statement, where "T50LINK.CMD" is an ASCII linker-command file that specifies all .OBJ files to be linked by the dsplnk.exe program, as well as specific information about memory addresses for some sections. Note that all executable code that is re-locatable starts at 400H. All code has been tested using version 6.40 of "dspa.exe" and "dsplnk.exe".

4) Copy the resulting "TMS32025.OUT" file to directory with Hypersignal software, run Hypersignal, and invoke the appropriate functions. Hypersignal reads the "COFF" file format directly, downloads all code and data, and begins execution at the reset vector (location zero).

MISCELLANEOUS NOTES ABOUT CHANGING TMS320C5x SOURCE CODE

1) If you need to change default buffer addresses, please consult the comments and suggestions in the T50DEF.ASM file.

2) The LNK.BAT file produces a "T50LINK.MAP" map file which you can use to check for overlaps and current locations of variables, routines, and code sections. The map file also shows the starting address and length of each module included in the current executable output file. As you add significant amounts of code, you should pay extra attention to the length of the executable file as compared the amount of program memory available. If you exceed the available program memory, problems can occur that seem to result from other causes, and which can lead you to spend unnecessary time searching for the real cause. See the "Onboard Memory Usage" section for individual boards in chapter 3 for hardware-specific memory amounts.

3) The T50LINK.CMD file creates an uninitialized "hole" from 26H through 3FFH that can be used for monitor and debugger programs. Because this section is uninitialized, the Hypersignal COFF loader will not write anything to this area.

4) If you need to use an external sampling clock, follow the basic guidelines below:

a. Set any jumpers required on the board itself. See the jumper list and/or hardware manufacturer documentation.

b. Enter '-EXT' after the board designator in the Hypersignal System Configuration menu. This causes the high-level software to set the value of the shared variable "FS_MODE" as appropriate for external clock operation. If no setting is possible due to the design of the board, then the value of FS_MODE will be zero. If the high-level software does not set the value correctly, or you introduce new external sampling circuitry to the hardware, we suggest that you continue to use the FS_MODE variable to control the hardware, by directly changing the DSP source code as appropriate.

c. If you need to know the actual sampling rate value, this is contained in the shared variable "FS_VALUE". The value is in Hz (Hertz); note that for TMS320C5x boards, which use 16-bit wide memory accesses, the maximum value that can be represented is limited to 65535. This value is initialized by the high-level software, based on what is entered in the current menu (SAMPLING FREQ field).

d. Enter SAMPLING FREQ field values in all high-level menus to match the actual external sampling clock rate, so that time and frequency unit readouts in all real-time and post-acquisition displays are accurate.

ACCESS AND CONTROL OF USER-DEFINED TMS320C5x SOURCE CODE FROM HYPERSIGNAL

The Hypersignal-Macro family of software packages and the Hypersignal macro language contain features and functions that facilitate access and control of user-defined DSP source code, such as:

1) The ability to specify different executable DSP files as the file to load and run when a DSP-based function or display is invoked from Hypersignal. In the System Configuration menu, entering a filename, such as “new.out”, after the board designator will override the Hypersignal default filename.

2) A macro language function (WriteDSPVal) that allows a macro to cause Hypersignal menu and internal values to be written to arbitrary locations in DSP memory.

3) A macro language function (WriteDSPMem) that allows a macro to cause arbitrary values to be written to arbitrary locations in DSP memory.

For more information about the latter two functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

Note that both the WriteDSPMem and WriteDSPVal functions always make full-word accesses, depending on the natural word-length of the DSP device. This can affect the address value used, if the DSP is capable of accessing data pieces smaller than full-word length. Also, note that both functions are capable of accessing all memory types supported by the DSP chip. For a complete list of internal Hypersignal values that can be written to arbitrary DSP source code memory, see Appendix A, WriteDSPVal Constants, and for fully detailed information about the syntax and usage of these functions, see Hypersignal Users Manual section 7.10, Macro Commands and Functions.

For TMS320C5x boards, you should avoid writing to program memory between 26H and 3FFH, which is typically left open for TMS320C5x debuggers, and 42DH and 4FFH, which is used by Hypersignal to store host-DSP communication variables and structures. Also, note that if Hypersignal DSP TMS320C5x code is allowed to run, it stores local variables in onchip data memory locations 0-127, location 2FF, and 896 through 1023 (filter coefficients and delay elements).

page left blank intentionally

Appendix A: WriteDSPVal Constants

Many internal Hypersignal values can be written to user-defined DSP source code from a macro language file using the WriteDSPVal function. This function has the basic form:

WriteDSPVal(memtype, addr, value)

where memtype is the type of memory (program, data, etc.), addr is the desired destination address, and value describes the data to be written. See Users Manual section 7.10, Macro Commands and Functions, for fully detailed information on the syntax and usage of this function. The possibilities for "value" in the above function call are listed below. Note that in some cases, a reserved word macro language constant is available that allows you to refer to the value in mnemonic fashion.

Value	Constant or Mnemonic Reference	Description
DSP/Analog Board	DSPBOARD	Indicates current board selection. For TMS320C25-based boards, the following values are possible: 0 = LSI/Spectrum, 1 = Ariel DSP-16, 2 = Texas Instruments SWDS, 3 = Intelligent Instrumentation PCI2020C. For DSP56001-based boards, the following values are possible: 0 = LSI/Spectrum, 1 = Motorola ADS56000 Development System with Ariel ADC56000 analog I/O, 2 = Ariel PC-56, 4 = Motorola ADS56000 Development System with DSP56ADC16 EVB analog I/O, 5 = Ariel DSP-56, Ariel PC-56D = 6, 7 = User-Defined, 8 = Techron TEF20HI. For DSP32C-based boards, the following values are possible: 0 = LSI/Spectrum, 1 = Ariel DSP-32C, 2 = AT&T Development System, 3 = CAC AC5 series, 4 = Intelligent Instrumentation ZPB34, 5 = MTT Lory DSP4200 System. For TMS320C30-based boards, the following values are possible: 0 = LSI/Spectrum, 1 = ASPI Banshee, 2 = Sonitech Spirit-30, 3 = Texas Instruments EVMC30, 4 = DSP Research TIGER 30. For 77220-based boards, the following values are possible: 0 = NEC DDK 77P220 Development System. For ADSP-2101-based boards, the following values are possible: 0 = LSI/Spectrum. For DSP96002-based boards, the following values are possible: 0 = LSI/Spectrum, 1 = Ariel DSP-96, 7 = User-Defined.

		For DSP3210-based boards, the following values are possible: 0 = Ariel DSP-3210.
Operating Mode	430H	indicates current operating mode (see above)
DSP Busy Flag	431H	indicates whether DSP is busy (1 = busy, 0 = not busy)
Overflow Flag	432H	indicates whether overflow has occurred in a DSP operation
Minimum |val|	433H	minimum absolute value in resulting DSP output buffer
FFT Order	DSPFFTORD	FFT order to be used in DSP function processing
Framesize	DSPFR	framsize to be used in DSP or math function processing
FFT Length	DSPFFTLEN	FFT length to be used in DSP function processing
Maximum |val|	437H	maximum absolute value in resulting DSP output buffer
Real/Imag Flag	438H	indicates whether DSP is required to produce real and imaginary output
Mask	439H	special-purpose analog I/O driver value currently used only by Intelligent Instrumentation PCI2020C and Techron TEF20HI
Offset	43AH	special-purpose analog I/O driver value currently used only by Intelligent Instrumentation PCI2020C
Sampling Rate Mode	DSPFSMODE	sampling rate control or mode register value for analog I/O subsection on DSP/Analog board (see above)
Trigger Value	43CH	trigger value for instrumentation functions (positive or negative, in A/D units)
Buffer Length	43DH	real-time processing framesize, or I/O buffer length
Current Host Buffer Flag	43EH	indicates current host I/O buffer (0 or 1)
Current DSP Buffer Flag	43FH	indicates current DSP I/O buffer (0 or 1)
Input Data Scale Factor	440H	scale factor to be multiplied with acquisition (input) analog I/O data (all traces)
Input Data Offset	441H	data offset to be added to acquisition (input) analog I/O data (all traces, in A/D units)
Window Scaling Factor	442H	window scaling factor to compensate for effects of currently applied FFT window

Phase Data Required	443H	indicates whether phase data is required in current operation
Trace Duplicate Flag	444H	indicates whether analog I/O data from trace 1 channel should be duplicated onto trace data
Filter 1 Coefficient Address*	DSPFILTADDR1	address to which to download filter 1 coefficients
Analog I/O Start Channel	DSPCHAN	starting channel in analog I/O physical channel list (see also Number of Analog I/O Channels, below)
Trigger Channel	447H	trigger channel for instrumentation operations
Output Data Scale Factor	448H	scale factor to be multiplied with generation (output) analog I/O data (all traces)
Output Data Offset	449H	data offset to be added to generation (output) analog I/O data (all traces, in A/D units)
Filter 2 Coefficient Address*	DSPFILTADDR2	address to which to download filter 1 coefficients
Real Data Maximum |val|	44BH	maximum absolute value in resulting DSP output real data buffer
Imaginary Data Maximum |val|	44CH	maximum absolute value in resulting DSP output imaginary data buffer
Filter 1 Length	DSPFILTLEN1	length of filter 1 coefficients (FIR), or number of biquad cascade sections (IIR)
Filter 2 Length	DSPFILTLEN2	length of filter 2 coefficients (FIR), or number of biquad cascade sections (IIR)
Log Flag A	44FH	indicates whether log processing is required in current mode for trace 1
Log Flag B	450H	indicates whether log processing is required in current mode for trace 1
Filter Type A	451H	filter type and structure for trace 1 data (0 = FIR, 1 = IIR structure type 2, 2 = IIR structure type 3; see Users Manual section 4.9.5, Generating DSP Code for Filters, for more information)
Sampling Rate Value	DSPFSVALUE	sampling rate value in Hertz
Power Spectra Flag A	453H	indicates whether power spectra averaging is enabled for trace 1
Power Spectra Flag B	454H	indicates whether power spectra averaging is enabled for trace 2
Power Spectra Coefficient A	455H	exponential averaging coefficient a in equation y[n]=ax[n]+by[n-1]

Power Spectra Coefficient B	456H	exponential averaging coefficient b in equation y[n]=ax[n]+by[n-1]
Transfer Function Flag A	457H	indicates whether transfer function processing is enabled for trace 1
Transfer Function Flag B	458H	indicates whether transfer function processing is enabled for trace 2
Maximum Amplitude A	459H	resulting maximum amplitude for trace 0 from acquisition operation
Maximum Amplitude B	45AH	resulting maximum amplitude for trace 1 from acquisition operation
Number of Analog I/O Channels	DSPNUMCHAN	number of analog I/O channels in operation
Board Subtype	45CH	board “subspecies”; used in cases of slight variations in DSP/Analog boards of the same basic model type, from the same manufacturer
Time Data Base Address*	DSPTIMDATAADDR	time domain analog I/O data or processing data base address
Real/Imag Data Base Address*	DSPRIDATAADDR	real and imaginary processing data base address
Magnitude/Phase Data Base Address*	DSPMAGDATAADDR	magnitude and phase processing data base address
Window Data Base Address*	DSPWNDDATAADDR	window (Hamming, Blackman, etc.) processing data base address
Monitor Mode	461H	a 1 enables “monitor mode” (digital loopback), a 0 disables
Module Indicator	462H	indicates current A/D module in a carrier-analog I/O daughterboard system (currently only used for Intelligent Instrumentation PCI202C system)
AIC Control Word / CODEC Control Word 1	DSPSERIALCTRL	AIC (analog interface chip) control word or CODEC control word 1 (boards with TLC3204x, T752x, CS42xx, and other programmable-chip analog I/O solutions utilize this value)
Log Coefficient 1	464H	coefficient c1 in DSP scaling equation y(x)=c1∗log10[x]+c2, where x is FFT output
Log Coefficient 2	465H	coefficient c2 in DSP scaling equation y(x)=c1∗log10[x]+c2, where x is FFT output
Acquisition Delay Counter	466H	value (number of samples) to delay acquisition after onset of output stimulus in stimulus & response operation
Stimulus Delay Counter	467H	value (number of samples) to delay stimulus output after acquisi-

		tion begins in stimulus & response operation
Stimulus Data Base Address*	DSPSTMDATAADDR	output stimulus data base address
Stimulus Data Buffer Length	469H	stimulus data framesize, or output buffer length
Filter Type B	46AH	filter type and structure for trace 2 data (0 = FIR, 1 = IIR structure type 2, 2 = IIR structure type 3, -1 = filter 2 identical to filter 1; see Users Manual section 4.9.5, Generating DSP Code for Filters, for more information)
Filter Quantization	46BH	filter quantization (in number of bits) for fixed-point DSP devices
Filter Coefficient Update Flag	46CH	flag indicating that DSP must update filter coefficients into onchip data RAM (only valid for devices with onchip RAM not accessible by the host interface)
CODEC Control Word 2	46DH	second CODEC control word (currently only used by some TMS320C30-based systems)
CODEC Control Word 3	46EH	third CODEC control word (currently only used by some TMS320C30-based systems)

*Values so marked indicate “indirect addresses”. The DSPWriteVal function simply replaces the current address with the new “addr” value specified in the DSPWriteVal function, instead of writing the current address to a new location. This causes Hypersignal to use the “addr” value specified in the DSPWriteVal function when downloading data, filter coefficients, etc. afterwards.

page left blank intentionally